University *of* Leicester

A Numerical Investigation of Time Resolved Flows Around Turbine Blades

By Edward J. Brooksbank

A thesis submitted to the University of Leicester
in partial fulfilment for the degree of
Doctor of Philosophy

Department of Engineering

Supervisor: Prof. J.P. Gostelow
Co-supervisor: Dr A. Rona

September 2001

A Numerical Investigation of Time Resolved Flows Around Turbine Blades

Edward J Brooksbank[†]

Department of Engineering, University of Leicester, Leicester, LE1 7RH, UK.

ABSTRACT

The complex nature of turbomachinery flows and the scale of associated flow phenomena such as shock waves and vortex shedding, apply constraints to the methods by which the flow can be analysed experimentally. Computational techniques have quite successfully been applied to the flow around turbine blades, but the transient and periodic phenomena observed in experimental studies have not been fully investigated. In this work an original working computational code is presented for time-resolved flows around turbine cascades. The code has been verified using test cases relevant to transonic flow. Some of the problems associated with computational techniques have been highlighted; these include the large number of schemes that are available, each with its own advantages and disadvantages. The code has been applied to a geometry representing highly loaded turbine blading currently under study at the National Research Council of Canada; this was also used extensively in previous computational and experimental investigations. The blading chosen has a relatively thick trailing edge, necessitated by cooling considerations. A distribution of the flow properties on the surface of the blade has been determined, from which an equivalent water table model has been designed based on the principle of the hydraulic analogy. The water table model thus generated represents a further method for experimentally investigating flow phenomena without the complexity of analysing very high frequency oscillations *in situ*. The time-resolved flow field has been computed showing unsteady phenomena. The unsteady phenomena have been shown to compare favourably with the unsteady features observed in preliminary experimental results. In the process, energy separation has been predicted to occur not only in the coupled wake region, but also for the first time within Kelvin-Helmholtz instabilities present in the trailing edge shear layers.

[†] email: ejb13@le.ac.uk

ACKNOWLEDGEMENTS

It is not practical to mention all those people who have helped in the production of this thesis, so I will try to restrict the list to its bare essentials.

I would initially like to thank Professor J.P. Gostelow of the University of Leicester, Engineering Department for the promotion of this thesis, his support for the duration of the project and his patience when progress was slow. Also thanks go to Dr Simon Hogg for initially acting as co-supervisor, and his continued support and input; Dr Aldo Rona subsequently acted as my co-supervisor.

Many thanks are due to Dr Rona for his large amount of input and deep understanding of computational fluid dynamics, without which some sections of the thesis may have been very thin.

I would like to thank my colleagues at the university for a very pleasurable time and a comfortable working environment, namely Pete, Denver, Ike, Paul, and Jon, you know who you are! The discussions within the group have been invaluable over the last few years.

I would also like to express my thanks and acknowledge the work of Jonathan Richard Shewchuk for his excellent triangulation software called 'Triangle' that has allowed me to concentrate more thoroughly on other areas of the research[‡], and Yousef Saad for the GMRES routines within his 'Sparskit' Fortran routines for solving sparse matrices[§].

Lastly, many thanks to my partner Sarah, for having infinite faith and patience whilst 'my writing-up will only take a few more weeks', and to my son, Jacob.

CONTENTS

NOMENCLATURE

A_n	area of cell n
$\widetilde{A}$	Roe Matrix
$\left|\widetilde{A}\right|$	Diagonalised Roe Matrix
CFL	Courant-Friedrichs-Lewy
d_n	number of incident vertices to node k
e_0	total energy per unit volume
h_0	total enthalpy
i	unit vector in the x direction
j	unit vector in the y direction
$\boldsymbol{E}$	Euler Flux Vector
$\boldsymbol{F_v}$	element averaged viscous flux term
$\boldsymbol{H}$	Source term
$\boldsymbol{H_i}$	Inviscid source term
$\boldsymbol{H_v}$	Viscous source term
k_c	thermal conductivity
M	Mach number
$\hat{n}$	unit vector in direction of centroid dual face
p	static pressure
$\dot{q}_x, \dot{q}_y$	x and y components of heat flux
$\boldsymbol{Q}$	vector of conserved variables
Q_i	element of $\boldsymbol{Q}$ vector
r	trailing edge radius
T, T^{-1}	diagonalising matrices
T_0	total temperature
t	time
u,v	velocities in x and y directions respectively
x,y	global co-ordinates
γ	specific heat ratio

λ_i	eigenvalue i
ρ	density
ε	entropy fix variable
μ	sum of the laminar and turbulent viscosities
μ_{lam}	laminar viscosity
μ_{turb}	turbulent viscosity
Ω_n	centroid dual control area for cell n
Ω	vorticity
Λ	diagonal eigenvalue matrix
Δu, Δv	change in velocity in x and y directions
Δp	pressure change
$\Delta \rho$	density change
ΔQ	change in conservative vector Q
$\widetilde{U}$	Roe-averaged velocity in direction of vector $\hat{n}$
ΔU	change in velocity in direction of vector $\hat{n}$
τ_{xx}, τ_{xy}, τ_{yy}	Viscous shear stresses
$\tau'_{xx}, \tau'_{yy}, \tau'_{\theta\theta}$	Viscous shear stresses associated with stream tube thickness

Subscripts

a	air
k	edge k
lam	laminar
max	maximum
min	minimum
n	node n
o	total (stagnation)
s	stagnation conditions
turb	turbulent
w	water
x,y	partial derivatives with respect to x and y
L	left side of dual face

R	right side of dual face
v	viscous terms
∞	free-stream value

Superscripts

~	Roe-averaged quantity
^	unit vector
*	non-dimensional quantity
'	shear stresses associated with stream tube thickness

1   INTRODUCTION

It has been known for many decades that the flow within turbomachines, such as steam and gas turbines, is highly complex in nature. Two phenomena that present themselves are shock waves, and energy separation in the trailing edge region of turbine blades. Through research and development over the years, turbines have become increasingly more efficient. This was originally due to parametric studies, but later, advancements in flow measurement techniques such as Schlieren, LDA (Laser Doppler Anemometry) and more recently PIV (Particle Image Velocimetry) have been utilised. These have been successfully combined with CFD (computational fluid dynamics), a numerical method for modelling fluid flow, to produce increasingly accurate results. In fact, since its development in the 1930s, the gas turbine has increased in power by a factor of 50, accompanied by a reduction in fuel consumption of 75%. The gas turbine industry has now become a dominant force in world markets, and the emphasis on producing efficient and cost effective turbomachinery is very high. Although turbine blading is reaching the limits of its efficiency in terms of current technology, there is still the potential for a few percent increase and the benefits of research into realising this increase are still viable. Furthermore, the cost and weight benefits of higher loading, and the accompanying reduction in blade count have become increasingly important.

Turbine blading[1] is the mechanism by which gas flow is redirected through a turbomachine; it is therefore important that the phenomena associated with the flow behind each blade are fully understood.

Two principal features are shock waves, discontinuities that form when supersonic airflow is reduced to subsonic velocities, and energy separation. Energy separation is the re-distribution of total temperature in the trailing edge of turbine blading, resulting in a cold region along the centre-line of the wake; this discovery was based on work originally conceived by Eckert (1943). Ryan (1951) postulated that the process of energy separation was associated with vortex shedding, the most common form of which is the Von Karman vortex street - where a series of vortices are alternately shed from either side of a blunt trailing edge. This was later to be proven by Kurosaka *et al.* (1987).

---

[1] Turbine blading: a series of identical turbine blades arranged radially about a central hub.

Due the tight confines of turbomachinery, physical analysis of these phenomena *in situ*, is usually very difficult. As a result, alternative methods for the investigation of trailing edge phenomena have recently been developed. Over the past few decades, as the power of computers has increased, the application of numerical solutions to the flow around turbine blades has increased. The complications described above make turbine blading an ideal case for modelling computationally. In recent years, computational work has been used quite successfully in the simulation of flow around turbomachinery, including the modelling of unsteady phenomena such as of vortex shedding.

1.1   Aims and Objectives

The aim of the thesis was to develop and verify a working numerical code, and use this to model a known turbine blade. The Mach number distribution for the turbine blade can then be used to design a water-table model based on the hydraulic analogy that will exhibit the same characteristics, but at a much lower shedding frequency. The water table model can then be used for more detailed analysis of the above mentioned flow phenomena.

For this research, a highly loaded turbine blade tested by the CNRC (Le Conseil National de Recherches du Canada - National Research Council of Canada) has been chosen. This particular blade has been used extensively in previous computational and experimental investigations, mainly due to its thick trailing edge. The purpose of the thick trailing edge was to allow for blade cooling; the cooling has not been taken into consideration for the current study. The flow region of interest is the transonic region defined by an inlet Mach number of 0.112, and an inlet to outlet pressure ratio of 2.3. This gives an exit Mach number of 1.16. The flow at the trailing edge of the blade is transonic giving the required periodic phenomena associated with the interaction of the shock waves and the vortex shedding. Computational work to date has already produced excellent predictions of Von Karman vortex shedding, Arnone & Pacciani (1997), and Carscallen *et al* (1998), but so far none has simulated periodic features associated with transonic flow in which various permutations of the traditional vortex shedding present themselves.

The first aim of the research is to develop a working numerical code to generate accurate Mach number distributions around the turbine blade. From these distributions, a water-table model can be designed according to the hydraulic analogy. The water table model is similar to a model presented by Roberts & Denton (1996), although the present model is designed for a supersonic exit Mach number. The water-table model can be seen in Figure 1.2, compared to

the equivalent turbine blading in Figure 1.1. The water table model exhibits advantages over the original turbine blade in that to give analogous characteristics to the turbine blade, flow of a much lower velocity is required. The flow velocities are now in the region of 0.44ms^{-1} as opposed to 300ms^{-1} for the turbine blade, thus the associated phenomena will be clearly visible to the naked eye; shock waves will be visible as stationary waves on the surface of the water and vortices will be present as small 'whirlpools' convecting downstream. Information may now be captured through simple techniques such as the use of video cameras and float gauges.

A RANS (Reynolds Averaged Navier-Stokes) finite volume code was developed for this study. Due to the very high Reynolds number of the flow under investigation, it was deemed that only the Euler solver would be required; the flow is predominantly Euler in the mainstream, with only a narrow boundary layer. The narrow boundary layer would have a minimal effect on the Mach number distributions around the turbine blade and the subsequent Mach number distributions around the water table model. This is justified by the work of Denton & Xu (1989). For complete accuracy, however, viscous computations would be required. These would have the added complication that minor discrepancies in the base pressure computations would affect the positioning of the shock waves in the trailing edge region, and the pressures over the back half of the turbine blade; Gostelow (1984).

The second aim of the research was to perform unsteady flow field calculations using the same numerical code and determine whether these calculations compare with previous experimental results, and preliminary experiments performed on the water table model.

The thesis is divided into the following chapters:

Chapter two: an introduction to turbine blading and previous computational and experimental work that has been performed.

Chapter three: an introduction to the computational fluid dynamics.

Chapters four through to nine: a description of the current solver that has been used for this study, including all the methods that have been tested.

Chapter ten: numerical validation of the solver.

Chapter eleven: a detailed description of the turbine blade under test.

Chapters twelve and thirteen: the hydraulic analogy and its application to the turbine blade and the water table geometry.

Chapter fourteen: conclusions and further work.
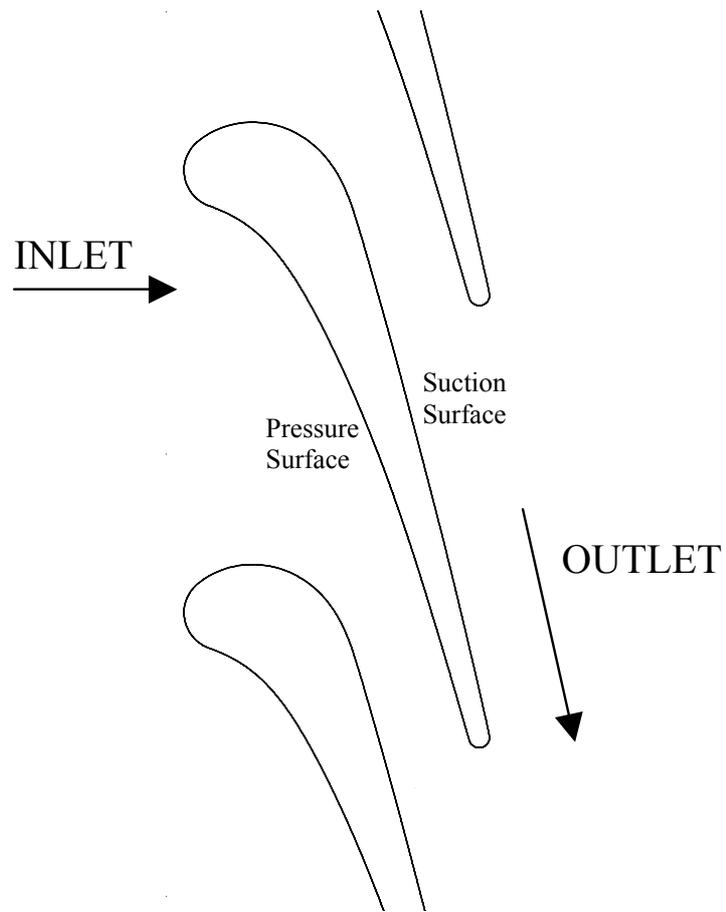
**Figure 1.1:** *Turbine Blading*
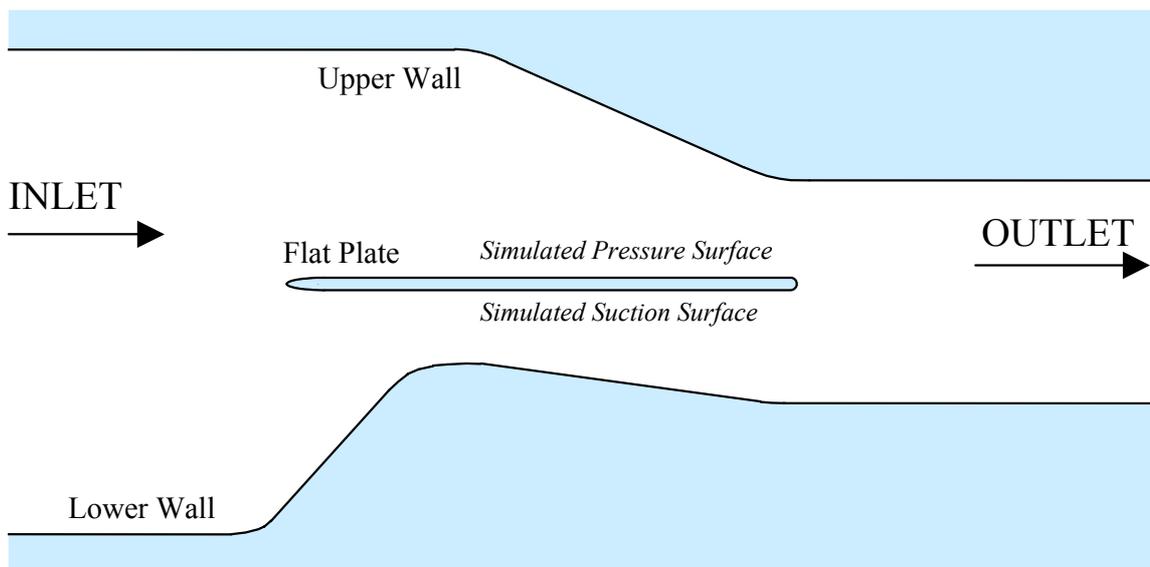
Figure 1.2: *The Water Table Model*

2   TURBINE BLADING

In turbomachinery, it is the turbine stage that is responsible for the conversion of thermal energy into mechanical energy. Examples of turbomachinery include the *gas turbine*, the *jet engine* and the *steam turbine*. The efficiency of the engine is limited by the need for very high temperatures in the combustion chamber and the early stages of the turbine. Continuous advances in heat-resistant materials, protective coatings such as ceramics, and cooling arrangements have made possible very high internal temperatures resulting in highly efficient turbomachinery. Improving the aerodynamic properties of the blades can further increase the efficiency. Better aerodynamic performance ensures that more energy is transferred into increasing the pressure through the compressor for example, rather than being expended in overcoming drag. Although these increases in performance will be in the order of a few percent or less, the economics of air transport and power generation ensure that research into these areas is still viable.

Due to physical constraints associated with turbomachinery, analysis of the flow conditions *in situ* can be very difficult. The gaps between consecutive rows of blades would be in the order of millimetres, and in the turbine stage, pressures and temperatures may be high enough to destroy most experimental probes. In fact, the gas temperatures in the turbine may be in the order of a thousand degrees, and it is only through effective cooling and design that the melting point of the blades is not exceeded. Other methods of investigating the flow through turbomachinery must therefore be developed that are not constrained by the factors described above.

2.1   The Turbine Cascade

The usual method of analysing turbine blades is with a *turbine cascade*. This is a series of blades, Figure 2.1, representing a small section of a circular turbine configuration - effectively unwrapping the blades to lie on a two-dimensional plane; Gostelow (1984) gives a highly in-depth study of this technique. Although the method has been used for the last seventy years, it is still a primary tool for the study of turbomachinery, and has by no means reached the end of its useful life. It is important to realise that the turbine cascade is merely a model and is a

method of simplification to result in a deeper understanding of the fundamental operation of the actual turbine blading.
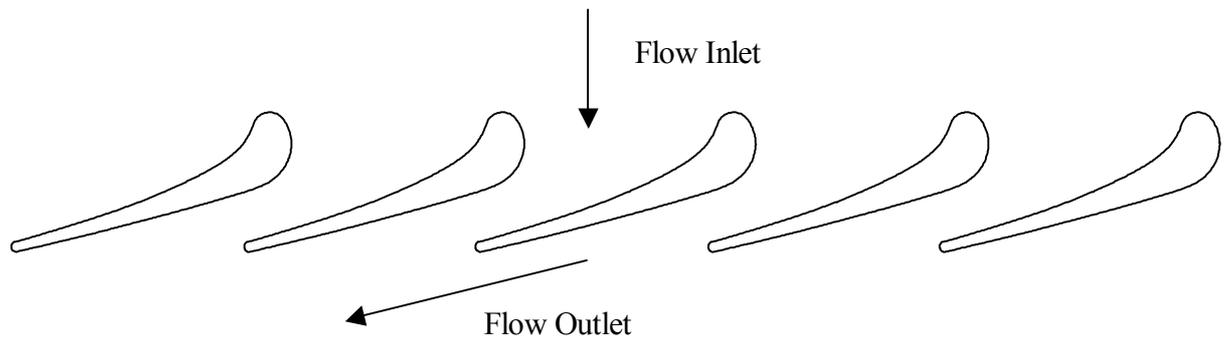
Figure 2.1: *Turbine Cascade*

Some approximations have to be made in the process of reducing the turbine disk to a turbine cascade; the turbine blade is a three-dimensional object as can be seen in Figure 2.2(a). As the tip of the rotor has a higher axial velocity, the blade needs to be twisted to ensure that the correct angle of incidence of the inlet flow is maintained along the leading edge. Consequently, the cross-section and camber at the tip of the blade do not match those at the root, Figure 2.2(b); the mean cross section or the cross-section at the midpoint is usually taken.
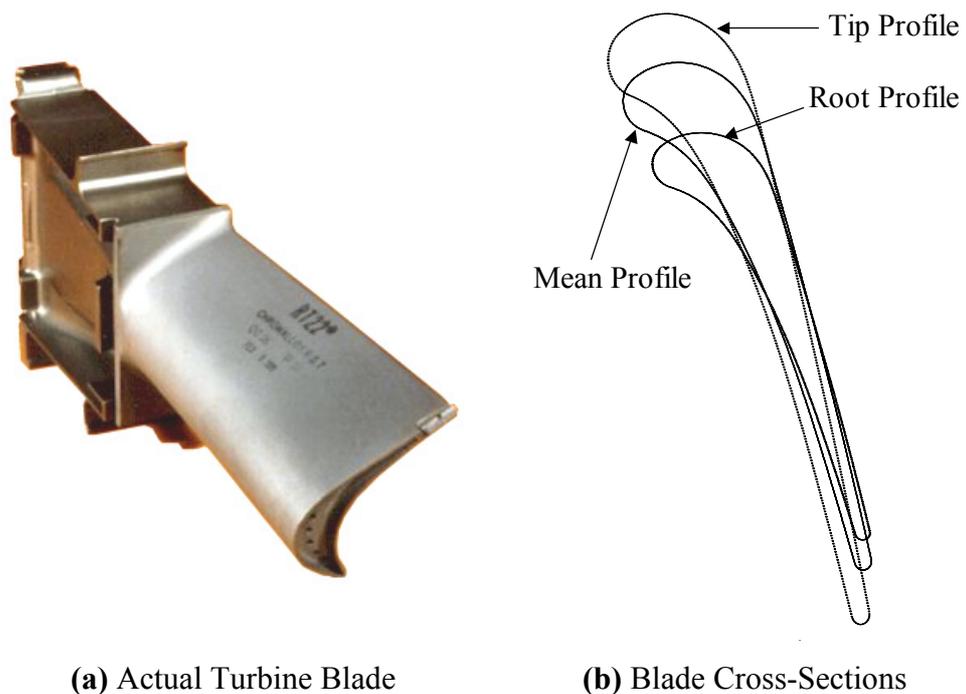
(a) Actual Turbine Blade   **(b)** Blade Cross-Sections

Figure 2.2: *Details of Turbine Blade*

The turbine cascade can now be integrated into an experimental rig that allows experimental probes and techniques to be used to their full potential. Techniques such as Schlieren, LDA (Laser Doppler Anemometry), PIV (Particle Image Velocimetry), and interferometry are just a few for which a great deal of information is available in current literature. Additionally, as the blade is not constrained to the limits of the turbine, it can be scaled to a size suitable for the analysis in question. This can be done confidently if the Reynolds number of the model matches that of the original.

2.2 Mach Number

Many phenomena that are present within turbomachines and the degree to which they interact are dependent on the Mach number of the flow. There are five main Mach number regimes into which the flow can be separated depending on the predominant flow characteristics, these can be seen in Table 2.1.

$Ma < 0.3$	Incompressible Flow Mach number effects are negligible. Viscous effects can now dominate the flow, particularly at very low Reynolds numbers.
$0.3 < Ma < 0.7$	Subsonic Flow Density effects are now present No shock waves within the flow
$0.7 < Ma < 1.2$	Transonic Flow Shock waves now begin to appear, mainly at the trailing edge of supersonic patches but as the Mach number increases to supersonic flow, weak shock waves also begin to appear.
$1.2 < Ma < 3.0$	Supersonic Flow There are now no subsonic regions within the flow.
$Ma > 3.0$	Hypersonic Flow Flow begins to depart from ideal gas theory with dissociation of molecules.

Table 2.1: *Mach Number Regimes*

2.3   Vortex Shedding

Numerous studies have been performed on a series of blade geometries exhibiting vortex-shedding characteristics. Figure 2.3 shows six blades that have been investigated under various loading conditions by: Heinemann & Butefish (1977), testing blades A-1 and A-2, Bryanston-Cross & Camus (1982), testing blades B-1 and B-2, a pair of high turning blades with slightly different rear suction surfaces, Sieverding & Heinemann (1989), testing blade C, and Carscallen & Gostelow (1994), testing blade D. Early tests by Carscallen & Oosthuizen (1989) in the trailing edge region of blade D indicated a very high redistribution of total temperature. Moustapha *et al.* (1993) compared experiment pressure profiles around the same blade, over a range of exit Mach numbers into the supersonic regime. Their work compared favourably with the computational work of Ni & Bogoian (1989), but no analysis of the wake flow was performed. The work does however provide good documentation for total pressure losses in the downstream flow field. Cicatelli & Sieverding (1996) performed low subsonic experiments at an exit Mach number of 0.4. The work showed vortex shedding in the wake with a distinct difference between the vortices shed from each side with the pressure side vortex being the more concentrated. This was also shown to be the case in the work of Roberts & Denton (1996). Although this was performed behind a flat plate, unequal pressures were generated on either side of the plate to simulate turbine blade trailing edge conditions.

The figure shows the range of Strouhal numbers observed for each of the blades depending on the exit Mach number. The Strouhal number is used to determine the vortex-shedding frequency and is defined as:

$$St = \frac{fD}{U_\infty} \tag{2.1}$$

where f is the shedding frequency, D is the trailing edge diameter and U_∞ the freestream velocity. A particular trend can be seen in Figure 2.3; in the subsonic region the Strouhal number decreases fairly uniformly followed by a large jump as the flow crosses into the transonic region.
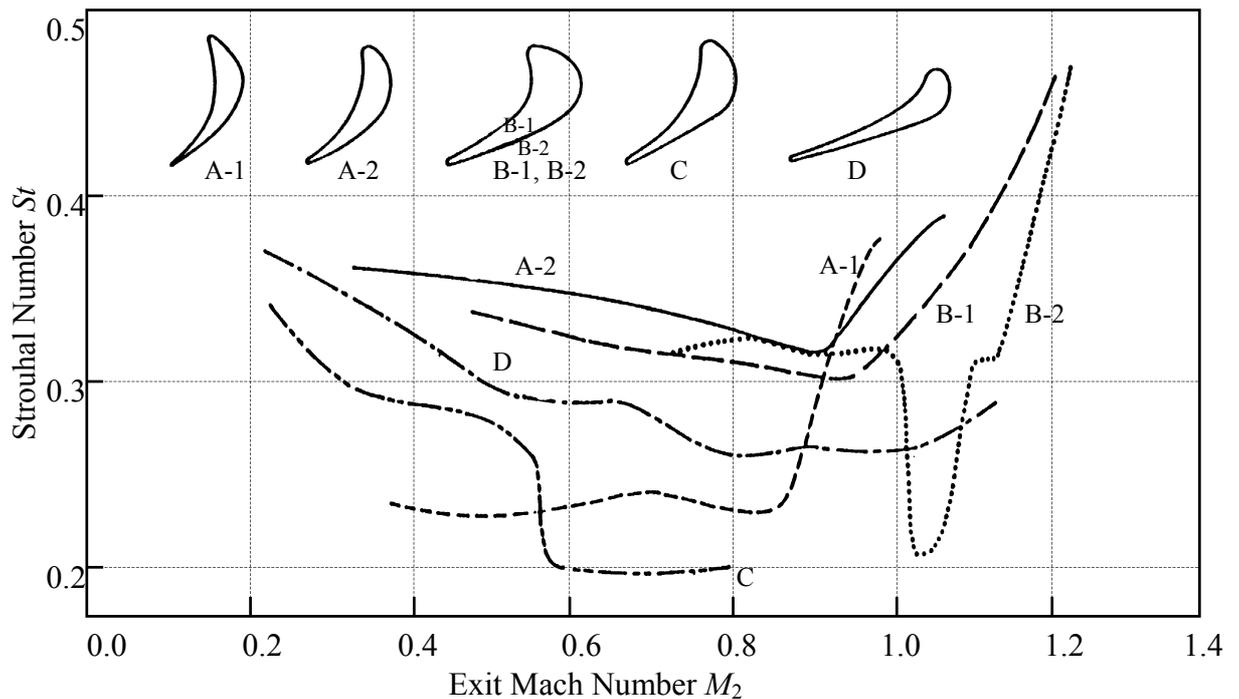
Figure 2.3: *Strouhal Number Based on Exit Mach Number, Cicatelli & Sieverding (1996)*

Following the work of Sieverding & Heinemann (1989), three boundary layer conditions can be defined based on the Strouhal number: $0.2 < St < 0.23$ indicates that the boundary layers are both turbulent, $St > 0.35$ indicates the boundary layers are both laminar, and intermediate values of St indicates that a transitional boundary layer state is present on one or both sides of the blade.

For the current study, blade D is being used. This particular blade has been used extensively in previous computational and experimental investigations mainly due its thick trailing edge, and is currently being investigated experimentally at the CNRC. The flow region of interest is the transonic region defined by an inlet Mach number of 0.112, and an inlet to outlet pressure ratio of 2.3. This gives an exit Mach number of 1.16. From Figure 2.3, this gives a Strouhal number of about 0.28.

2.4   Supersonic Flow Structure

The flow structure in the trailing edge region of the turbine blade changes radically when the flow crosses into the transonic regime. The wake structure no longer stems from the actual edge of the blade, but from a new point further downstream. This particular flow structure can be seen in Figure 2.4(a).

(a) Supersonic Trailing Edge   **(b)** Transonic Trailing Edge
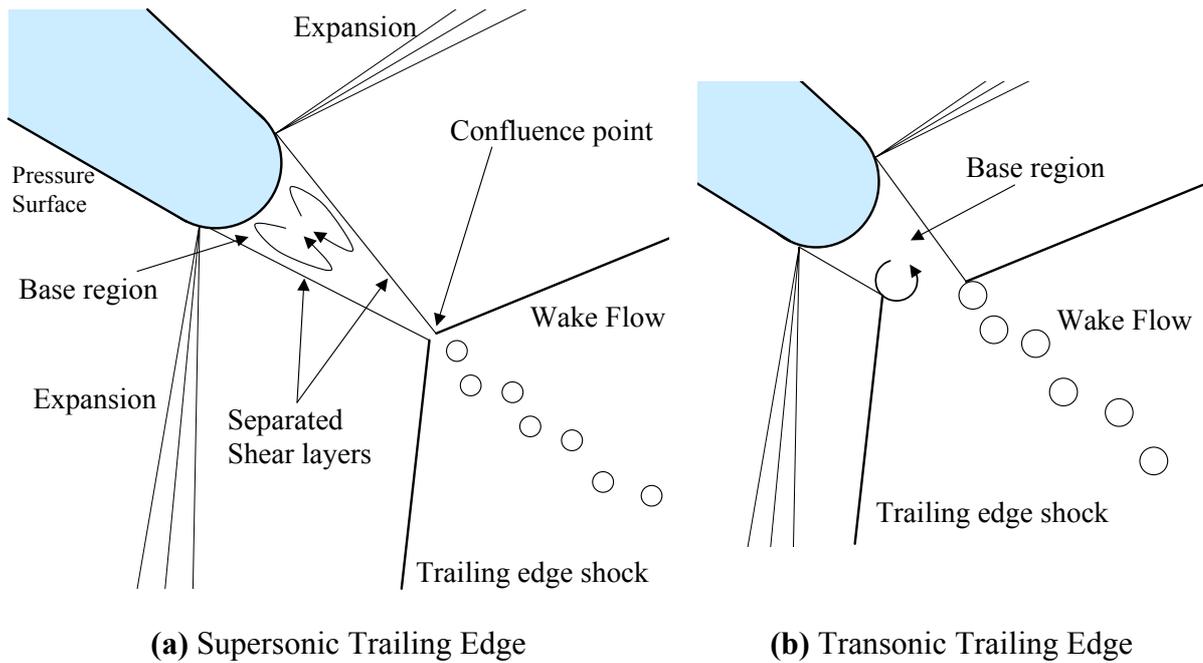
Figure 2.4: *Flow Structure for Trailing Edge*

This flow structure is now well understood, Denton & Xu (1989). The upstream flow on either side of the blade is supersonic. The base region is roughly triangular in shape, bounded by two upper and lower shear layers. The shear layers start where the supersonic flow separates from the body of the blade at the first point of curvature. The resulting change in flow direction generates expansion waves that propagate into the main flow. The amount of turning is related to the pressure in the base region. The flows from either side of the blade meet at a point downstream, termed the confluence point. The flows turn to a common direction at this point and a compression shock is formed. One side of this shock propagates downstream away from the blades, and the other impinges on the suction surface of the adjacent lower blade. The flow downstream of this trailing edge shock is supersonic.

There is also an intermediate transonic state in which the triangular base region is not fully developed, Figure 2.4(b). The triangular region is effectively truncated with no defined confluence point. Vortices are shed from the trailing edge of the blade as with the subsonic case, but are now confined to the channel defined by the shock waves that are formed where the flow is redirected by the vortices.

2.5   Computational Investigations

Over the past few years several studies have been performed on the flow through turbines. Xu & Denton (1987) compared experimental and computation work for four blades including blades B1 and B2 over a range of Mach numbers. Their work was Euler in nature, and generated good results, with a base pressure coefficients from the time marching code closely matching the experimental results. No details however were recorded with reference to the presence of vortex shedding. They concluded that the base pressure loss was proportional to the trailing edge thickness. However their later work, Xu & Denton (1990) gave more accurate base pressure predictions and implied that an Euler solver may be more confidently used for supersonic computations. Later work by Furukawa *et al.* (1992) on blade A-2 used a fully viscous implicit code. Their work showed a good agreement between numerical and experiment surface pressure distributions for an exit Mach number of 0.975. Vortex shedding was present although not very well resolved, and the Strouhal number was underestimated. This is probably due to the low density of the grid, with only sixteen points around the trailing edge of the blade. Arnone & Pacciani (1997) performed a computational study at an exit Mach number of 0.4, based on the blade tested at the VKI, as investigated by Cicatelli & Sieverding, (1996). Their work was of a very high standard clearly showing vortex shedding in the trailing edge region. The frequency of the vortex shedding was found to agree with the experimental work but overpredicted pressure oscillations. Carscallen *et al.* (1999) performed numerical and experimental work over a range of exit Mach numbers on blade D. Again they showed vortex shedding in the wake region, with a maximum reduction in wake temperature at an exit Mach number of 0.95. Both this work and the work of Arnone & Pacciani (1997) showed the effect the different boundary layer thicknesses had on the vortices being shed from either side of the blade. In particular, the vortex shed from the pressure surface is shown to be the more concentrated vortex with a more smeared vortex emanating from the suction surface. This corroborates well with the experimental work described previously. The computational work of Gehrer *et al.* (2000) was performed at exit Mach numbers of 0.62 and 0.7. The calculations were of a high quality and vortex shedding was clearly evident. This was performed with the VKI blade and computational results were in good agreement with their experimental work

Of particular interest is the work of Currie & Carscallen (1998), again on blade D. Results were generated for exit Mach numbers of 1.0 and 1.16, the latter matching the current study. Their computational results were in excellent agreement with experiment work clearly

showing vortex shedding and the shock waves forming at the confluence of the two shear layers. The numerical vortex shedding was in the form of the standard von Karman vortex street, the dominant shedding mode for this Mach number. Other vortex shedding modes are described in more detail in section 2.8.

2.6   Energy Separation

Energy Separation is defined as the rearrangement of total temperature in the wake of a bluff body into regions of higher and lower total temperature. In 1943, Eckert performed experiments measuring the surface temperature of thermally insulated cylinders in a cross flow of air, primarily at Mach 0.685. His findings were that the temperature at the upstream stagnation point was the expected temperature but at the trailing edge stagnation point temperatures as much as 20°C lower than expected values were measured. In terms of a recovery factor, defined as:

$$R = \frac{T(\theta) - T_\infty}{T_0 - T_\infty}$$

(2.2)

where T_∞ is the upstream static temperature, T_0 is the upstream total temperature and $T(\theta)$ is the total temperature at an angle of θ around the cylinder measured from the leading stagnation point (θ=0°), this gave negative values. For the leading edge R would be 1, and for the rear stagnation point (θ=180°), a value of about –0.1 was measured, in contrast to expected values of about 0.84-0.9 that had been measured in flat-plate experiments. The lower the value of R, the lower the temperature. These experiments were performed with a thermocouple embedded in a hollow rubber cylinder.

This startling discovery was later to be known as the 'Eckert-Weiss Effect'. It should be noted that the experiments were performed using a thermocouple, which only measured time-averaged temperature, so the 'Eckert-Weiss Effect' should be recognised as the time-averaged total temperature loss on the trailing edge of a cylinder. Although originally ridiculed by Prandtl, who stated that this temperature loss was due to experimental errors, Eckert's work was subsequently corroborated by Ryan in 1951. Ryan also extended this with the observation that the temperature loss was also present in the wake flow behind the cylinder, as can be seen in Figure 2.5, and not just on the surface.

Non-dimensional T_0
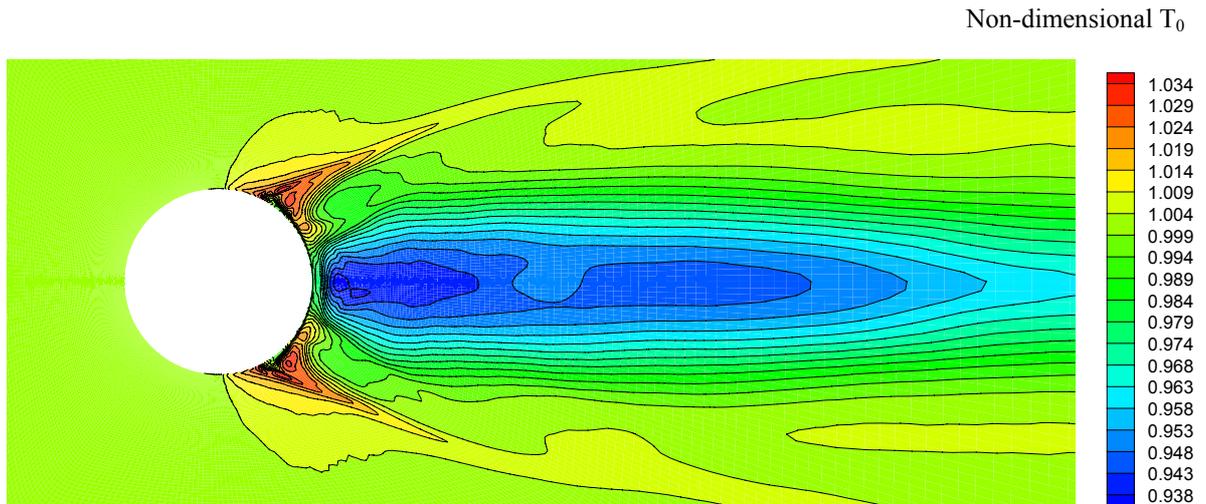
Figure 2.5: *Time Averaged Total Temperature[2]*

These results are particularly interesting in that they seem to contradict the law of conservation of total energy. In a steady state system, for adiabatic, isentropic flow, total enthalpy is conserved along a streamline:

$$h_0 = C_p T_0 = C_p T + \frac{U^2}{2} = \text{constant} \qquad (2.3)$$

As a consequence there should be no redistribution of total temperature. The above findings would imply that there was an unsteady effect performing work in the system. If the flow is indeed unsteady then the rate of change of total enthalpy along a streamline can be defined by:

$$\frac{dh_0}{dt} = \frac{1}{\rho} \frac{\partial p}{\partial t} \qquad (2.4)$$

This can be interpreted as pressure changes in the system perform work on a streamline resulting in a change in total enthalpy.

Ryan also noted that a particular whistling sound could be heard that coincided with the drop in the recovery factor. The sound heard was at a frequency matching that of the vortices being shed from the cylinder. He therefore posited that vortex shedding was the unsteady

---

[2] Courtesy: William Paul Bennett, University of Leicester.

mechanism responsible for the lower recovery factor. Thomann (1959) added further to this by his finding that suppressing the creation of the vortex shedding led to an increase in the recovery factor.

At around the same time another interesting discovery was made by G.J.Ranque, this effect became more widely known in the United States of America after 1947 through a small device now called the *Hilsch Tube* (Eckert, 1986). The tube consists of an inlet nozzle A, and a small orifice at C to let the central air travel through to the right. Compressed air is injected through the small nozzle at the tube wall A and tangential to the wall. Part of the flow then leaves through the orifice at C and the rest leaves through the peripheral opening at B. The interesting observation that can be made is that the total temperature at the cold outlet is significantly less than the total temperature of the inlet air, and the total temperature at the hot outlet is higher than the inlet air.
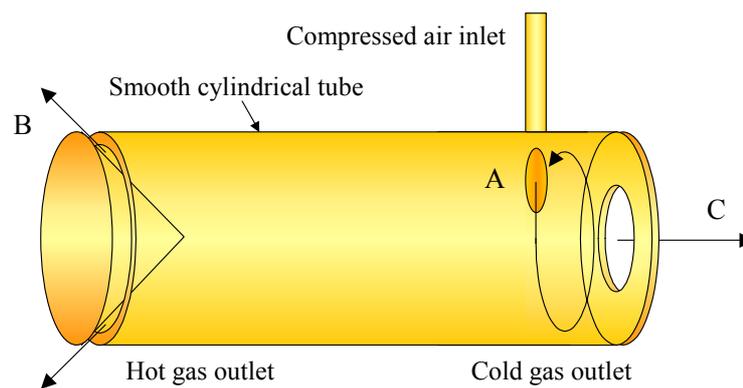
Figure 2.6: *Counter-Flow Hilsch Tube*

This effective separation of the inlet air into hot and cold streams is called the *Ranque Effect*. The flow inside the Hilsch tube is complex but well documented, Gutsol (1997), although the exact mechanisms of the temperature separation are not fully understood.

The flow can be described as similar to that found inside a cyclonic dust separator. This can be seen in Figure 2.7; compressed air enters through the inlet at A and forms a vortex around the periphery of the tube. The swirling flow progresses to the left hand end of the tube where the outer hot gas can exit through the ring orifice at a high pressure. The inner layers of fluid are redirected back to the cold outlet and form an inner vortex flowing to the right. To ensure that fluid does exit from the cold outlet, the ring orifice at the hot outlet to the left must be

closed sufficiently to give enough back-pressure. The flow in the Hilsch tube is known to be turbulent and particularly noisy during operation.
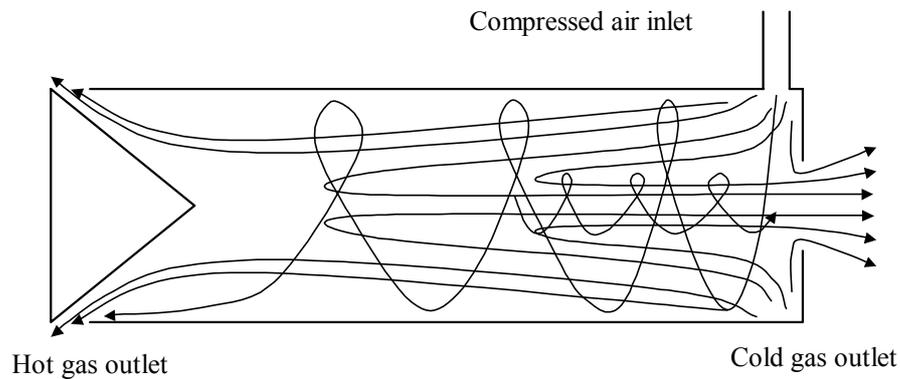
Figure 2.7: *Flow Pattern through Counter-Flow Hilsch Tube*

Recent theories posit that there are three effects responsible for the temperature separation: 1) Centrifugal forces; in the rotating column of fluid, centrifugal forces generate a pressure distribution across the tube from low pressure at the centre to high pressure at the outer edge. The lower pressure results in a lower temperature. 2) Shear stresses within the fluid; for angular momentum to be conserved, the inner layers of fluid must rotate faster, however, viscous forces between layers act to decrease the velocity of inner layers and increase the velocity of the outer layers. Consequently heat is transferred from the inner layer to the outer layer. 3) Turbulent mixing; particles of fluid move between adjacent layers and in doing so move into areas of lower or higher pressure. A particle moving towards the centre moves into a region of lower pressure, it can be considered to undergo adiabatic expansion with an associated decrease in temperature, and can now absorb heat from the surrounding medium. Conversely, a particle moving into a region of higher pressure can be considered to undergo adiabatic compression with an associated increase in temperature, and heat can be expelled to the surrounding medium.

With Eckert having renewed interest in this phenomena, Kurosaka *et al* (1987), began investigating the flow behind cylinders both experimentally and computationally. His experiments employed a technique of acoustic resonance to establish the extent to which vortex shedding was responsible for the energy separation. Defining the base pressure coefficient as:

$$C_{pb} = \frac{p_b - p_\infty}{\frac{1}{2}\rho_\infty U_\infty^2}$$

<div align="right">(2.5)</div>

where p_∞ is the upstream static pressure, ρ_∞ the upstream static density, U_∞ the inlet velocity, and p_b the pressure measured at the base of the cylinder; θ=180°. Kurosaka *et al*, measured simultaneous drops in C_{pb} and R. His conclusions were indeed that the Eckert-Weiss effect was a consequence of vortex shedding. Furthermore, his computational work also noted the temporary presence of hot and cold spots within the wake. He provided the following explanation for the mechanism by considering a fictitious particle moving along a streamline into the wake of the cylinder. The path of the particle will follow a *trochoidal* streamline. An easy way to visualize this is to imagine the locus that a point in the rim of a bicycle wheel will follow as it moves along the ground. The figure below shows a *cycloidal* streamline, the particular case where $U_v = U_\theta$
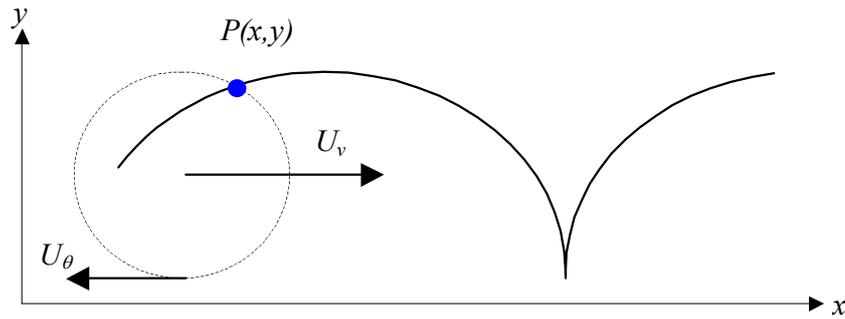
Figure 2.8: *Cycloidal Locus of Particle P, $U_v/U_\theta = 1$*

The above diagram shows a particle following a cycloidal streamline on a vortex being convected to the right. Now, from equation (2.4), for a constant C_p, the unsteady flow equation becomes:

$$C_p \frac{dT_0}{dt} = \frac{1}{\rho}\frac{\partial p}{\partial t}$$

<div align="right">(2.6)</div>

It can be seen that the total temperature of a particle changes with $\partial p/\partial t$. As the particle moves from the top of the vortex to the bottom, the vortex centre is approaching the particle resulting in $\partial p/\partial t$ <0, and so the total temperature must also drop. As the particle then moves

back to the top on the left of the vortex, $\partial p/\partial t > 0$ as the vortex centre recedes, and the total temperature increases. The net effect is that of a heat pump with heat being absorbed from the lower fluid and redistributed to the upper fluid. Near the centre of the vortex, $\partial p/\partial t$ is greatest, the largest temperature differences can be seen. As vortices are shed from either side of the cylinder, the vortex spans from the outer point of the cylinder to the midpoint, the centreline of the wake would then be the coolest being affected by both the upper and lower vortices. Figure 2.9 shows one of the computational results generated by Kurosaka *et al.* (1987), showing the cool wake and the warmer outer region. Although the cold spots are not immediately obvious, the hot spot stands out very clearly.

These results were further verified by Ng *et al* (1990) performing experiments in the wake of a cylinder with a high frequency aspirating probe. These experiments were performed at particular Mach numbers of the flow around a cylinder. Fox *et al* (1993) performed experiments investigating the temperature separation found in structures occuring at the interface between a jet and the surrounding fluid. Kurosaka (1992), working with O'Callaghan, measured a similar effect in the shear layer in a channel and also concluding that the magnitude of the temperature separation was approximately proportional to the square of the difference in Mach numbers of the two fluids. Carscallen & Oosthuizen (1989) obtained time-averaged total temperature and pressure downstream of a transonic turbine blade observing similar effects to those described above; in addition to the decreased total temperature in the centre of the wake there was an increase in the total temperature at the edges of the wake.
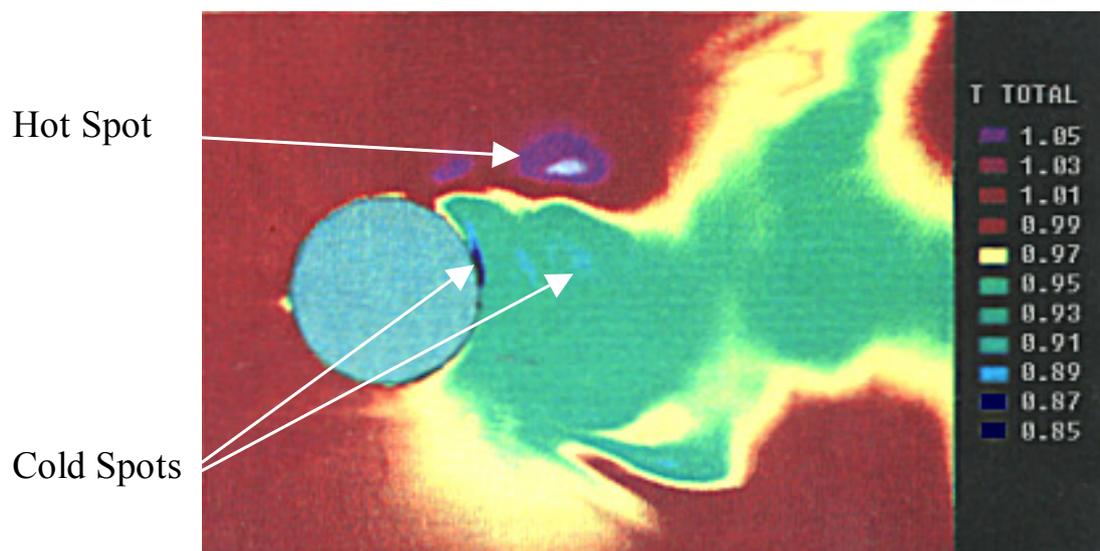
Figure 2.9: *Instantaneous Total Temperature, Kurosaka et al. (1987)*

Similarities can be seen between the flow in the Hilsch tube and a vortex at the point at which it is being shed from the trailing edge of a turbine blade. Both exhibit similar flow patterns and heat transfer characteristics.

2.7   Shock Waves

Weak pressure disturbances in a fluid are propagated through the fluid at a velocity defined by its speed of sound, a function of the properties of the fluid. Thus for stationary flow, pressure disturbances will radiate outward from a body in concentric circles, Figure 2.10(a). If the disturbance occurs at time t, then the figure shows the position of the wave at times $t+1$, $t+2$, and $t+3$, with the distance between waves being uniform. Shock waves are features formed in a fluid associated with the superimposition of weak pressure waves propagating through the fluid. For a body moving slower than the velocity of propagation, no shock waves are formed, as the pressure waves radiate from the body in all directions; this flow is termed *subsonic*, and is similar to the stationary case. For a body moving at the velocity of propagation, termed *sonic* or *transonic*, a normal shock wave is formed at the upstream point of the body as the leading edge of the waves coincide. If the body is moving faster that the velocity of wave propagation, then an *oblique* shock wave is formed and the velocity of the body is termed *supersonic*, and a *Mach cone* is formed.

(a) Subsonic $U = 0$ **(b)** Sonic $U = a$ **(c)** Supersonic $U > a$
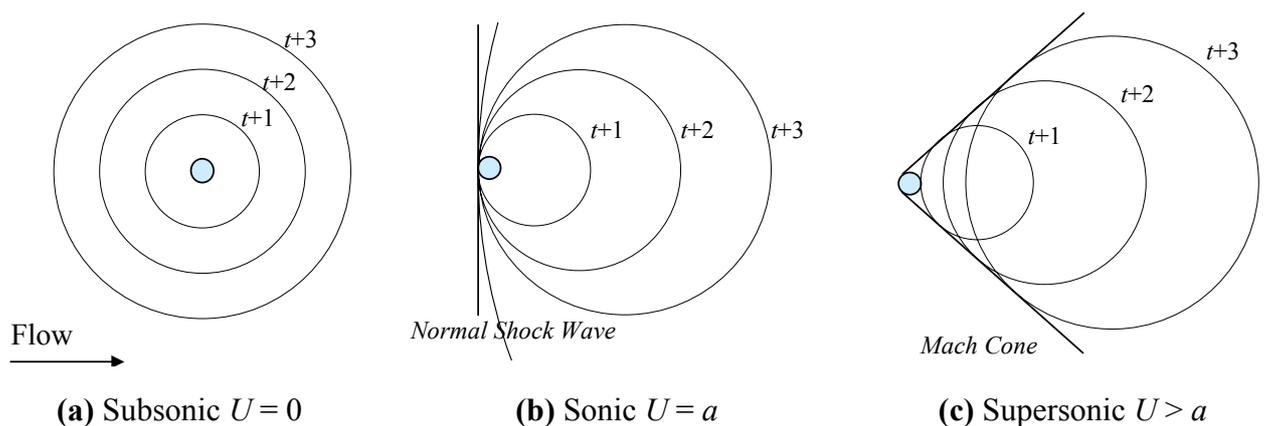
Figure 2.10: *Formation of Shock Waves*

Shock waves are one of the main causes for inefficiencies and energy loss within turbomachinery, Xu & Denton (1989). Energy losses that are associated with shock waves are due to heat conduction and viscous effects causing an increase in entropy.

2.8   Transient Vortex Shedding

Carscallen & Gostelow (1994), and Carscallen *et al.* (1996) performed experiments over a wide range of Mach numbers in order to demonstrate the existence of vortex shedding behind turbine blades. Their results showed this to be true, with vortex shedding occuring over an exit Mach number range of 0.5 to 1.17. Their experiments were performed using high-speed Schlieren. In the low Mach number range, the shedding was seen to follow the traditional Von Karman vortex street, but at transonic and supersonic exit Mach numbers, the vortex street became unstable and a series of transient features presented themselves. Several different patterns were observed and labelled, and for an exit Mach number of 1.16, they measured the percentage occurance for each pattern.

Shedding Pattern	Visual representation	Percentage occurance
No definable pattern	n/a	35.5
Classic Von Karman street	○ ○ ○ ○ ○ ○	30.5
Leaning Von Karman street	○○○○ ○○○○	16.0
No coherent structure	n/a	9.0
Doublets	○○ ○○ ○○ ○○ ○○ ○○	4.0
Couples	○○○○ ○○○○	3.5
Hybrid	○○ ○○ ○○ ○ ○ ○	1.5
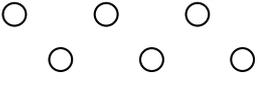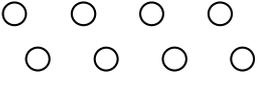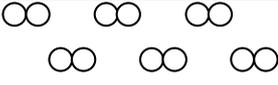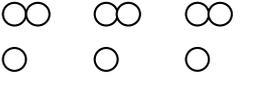
Table 2.2: *Vortex Shedding Patterns*

The above shedding patterns are intermittent, interspersed with regions in which there is no structure to the wake flow. This change in structure is accompanied by a change in base pressure, and is attributed to the relocation of the vortex shedding from the trailing edge to the point of interaction between the shear layers as described in section 2.4.

2.9   Concluding Remarks

A review of the present state of experimental and computational work around turbine blades has been presented in this chapter. This work is justified by the physical constraints of turbomachinery. The interaction between vortex shedding and shock waves in the trailing edge region of turbine blades has only been adequately explored in the last few years, and it is in the transonic regime that the unsteady phenomena and interaction are most dominant. The computational work investigated has shown that results of sufficient accuracy can be obtained, and can be used quantitatively in conjunction with experimental work. The work presented in this thesis offers an extension to this research.

It can be concluded from the work of Xu & Denton (1987,1990) that only an Euler computational solver is necessary for the calculation of accurate pressure distributions around turbine blades. The turbine blade in the current study is being tested at a high Reynolds number, making the flow predominantly Euler in nature. The areas in which viscosity becomes important are the boundary layers, the trailing edge region and the region of vortex shedding. The mechanism of vortex shedding is also predominantly Euler; the effect of viscosity is to smear the vortices. The Euler equations are well defined and non-problematic in their implementation, whereas the Navier-Stokes equations are more complex, requiring additional terms and equations to accommodate a relevant turbulence model. For the current test case, the solution to the numerical equations is very sensitive to the state of the boundary layers; the pressure surface boundary layer is assumed to be laminar, and the suction surface ranges from laminar to transitional, and then to turbulent towards the trailing edge. The turbulence model must be able to cope with these states otherwise accuracy no greater than the Euler equations can be achieved. Furthermore, correct representation of the base pressure requires extreme accuracy in the computation of the triangular base region; a highly accurate turbulence model would be required, but the results would be very sensitive to the turbulence model in question; Gostelow (1984). Consequently, viscous calculations are unlikely to produce a more accurate flow-field prediction: Furukawa *et al* (1992), and Cicatelli & Sieverding (1996). Extension to the Navier-Stokes equations would also require the definition of very dense meshes, particularly in the boundary layer regions, if the boundary layer structure is to be correctly captured this would increase the requirements for memory and processing.

3   COMPUTATIONAL FLUID DYNAMICS

With the advent of increased computer resources, Computational Fluid Dynamics techniques have become more accessible for the solution of non-linear differential equations such as the Euler and Navier-Stokes equations. Algorithms are now available that can solve the equations to machine accuracy and various acceleration techniques are available so that solutions can be generated within acceptable time frames. Such algorithms are now becoming a standard part of the design procedure for many engineering applications such as those of the turbine blade industry. Before any such calculations can be performed, an appropriate grid must be generated which represents the computational domain of interest. In past years numerical methods utilised *Structured* meshes. In recent years however, more complex techniques such as *Body Fitted Co-ordinates* and *Unstructured* meshes have been developed. An in-depth study of structured and body-fitted co-ordinate systems can be found in Hoffman & Chiang (1993). The most common methods of discretisation for engineering applications are grid based methods employing *finite element*[3] or *finite volume*[4] techniques. There are however spectral methods that do not require an underlying grid; these are not considered in the scope of this thesis.

3.1   Structured Grids

Structured meshes are comprised of a uniform assembly of polygons, the most common form of which is an assembly of regular quadrilaterals. All cells are known to have a fixed number of neighbouring cells that are arranged in a Cartesian co-ordinate system. Cells are numbered consecutively along the orthogonal axes of the co-ordinate system. Adjacent cells can then be immediately identified. The information for the variables can be stored at either the centre of the cells or at the vertices.

---

[3] Finite Element: An approximate method for solving partial differential equations by replacing continuous functions by piecewise polynomial approximations defined over polygons. This reduces the problem of finding the solution at the vertices of the polygons to that of solving a set of linear equations.

[4] Finite Volume: The integral form of the conservative laws is discretised directly over a control volume.

3.2   Body Fitted Co-ordinates

Body Fitted Co-ordinates are an extension of the structured system in which the required co-ordinates are conformally mapped onto a regular computational domain. This is achieved by considering a curvi-linear co-ordinate system (ξ,η) as opposed to a cartesian co-ordinate system (x,y). Co-ordinate transformation laws $\xi = \xi(x,y)$ and $\eta = \eta(x,y)$ generate a mapping from the physical domain (x,y) onto the computational domain (ξ,η). The structured method can then be used to generate a solution that is then inversely mapped back onto the physical domain to give the required result.
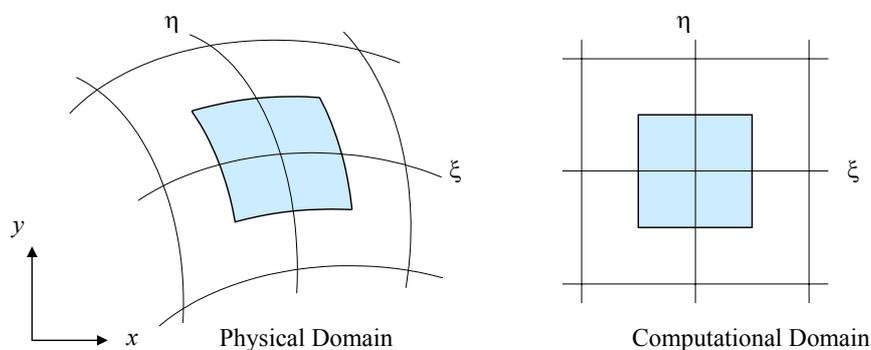
Figure 2.1: *Mapping of Physical Domain onto Computational Domain, Hirsch (1990)*

All of the major discretisation methods available can be performed on the above two methods without too much difficulty and such techniques can be implemented in a computationally efficient manner. However, a disadvantage of the above two techniques is that they do not lend themselves to the solution of highly complex geometries. In other words it may not always be possible to generate an acceptable grid. One solution to this is to sub-divide the domain into smaller regions and generate a grid for each region, a technique known as *multi-blocking* or *domain decomposition*. The blocks are then re-combined to give the required global solution. Structured grids can suffer from highly stretched cells when trying to model curved surfaces. The stretched cells have disproportionate dimensions with one axis being several times larger than the other. If the stretching occurs perpendicular to the flow direction, these cells can introduce numerical inaccuracies into the solution. However, stretching in the direction of flow in the boundary layers for example is very desirable.

3.3   Unstructured Grids

An unstructured grid differs from the above two methods in that there is no inherent structure based on an underlying co-ordinate system. They generally contain quadrilateral and triangular cells in two dimensions, and cuboid or tetrahedral cells in three dimensions, but other polygons may be used as required. A feature of such a grid is that each cell does not necessarily have a constant number of neighbours. A map of the connectivity of the vertices of the triangles is now used to determine adjacent cells. Unstructured grids are typically generated using *Delaunay* triangulation based techniques.

There is no concept of directionality to the grid; therefore methods applicable to a structured grid system cannot be used. Discretisation methods based on integral procedures, such as *finite element* and *finite volume* are found to be more suitable.

Unstructured grids therefore offer the advantage of discretising complex domains. They do however, require far more in terms of computer resources such as processing power and memory on a point-for-point basis. For a given number of points available to either the structured or the unstructured methods, the increase in resources is outweighed by the fact that gridpoints can be placed anywhere in the domain in the unstructured grid to give resolution where it is required.

(a) Structured Grid   (b) Body Fitted Grid   (c) Unstructured Grid
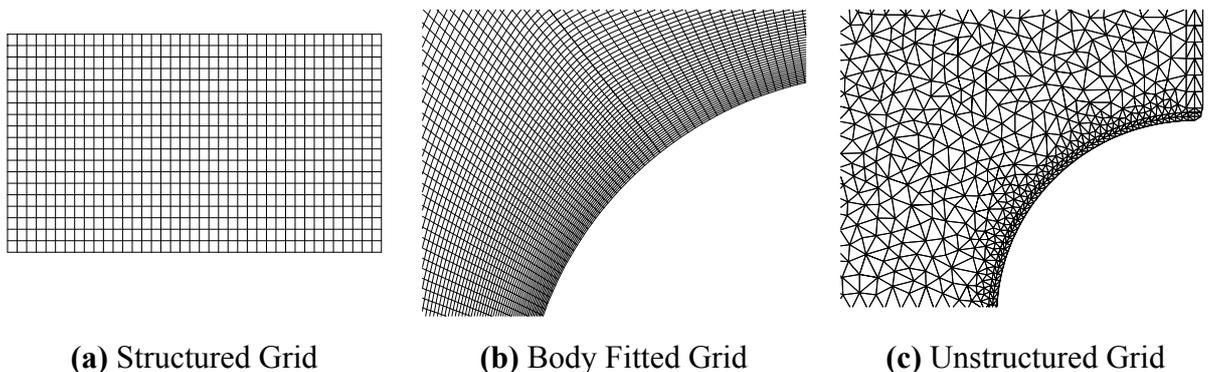
Figure 3.2: *Grid Types*

3.4   Delaunay Triangulation

Delaunay triangulation is directly related to the Dirichlet Tessellation, Bowyer (1981) of a set of points. A Dirichlet Tessellation of a point set is a pattern of convex regions, each being closer to some point than any others. A Voronoi diagram shows these regions; Figure 3.3:
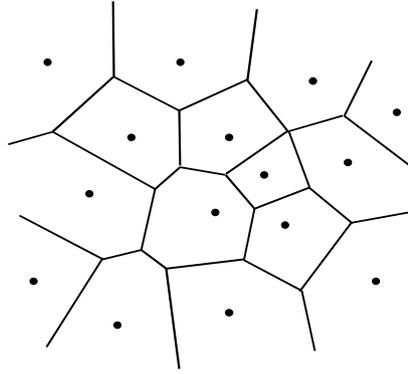
Figure 3.3: *Voronoi Diagram*

A Delaunay triangulation is the dual of the Voronoi diagram and is formed by connecting two points of the set if and only if their Voronoi regions have a common border section. The vertices of the Voronoi diagram will be the *circumcentres*[5] of the triangulation because each vertex will be equidistant from its neighbouring points. The Delaunay triangulation of a given point set will always be unique.

Incircle test TRUE   Incircle test FALSE
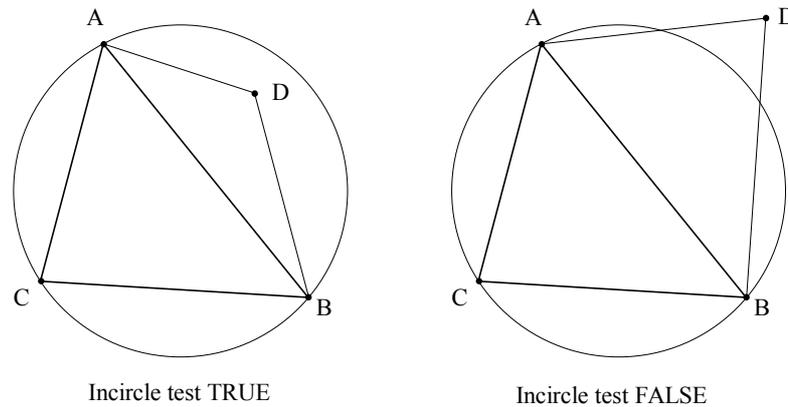
Figure 3.4: *Incircle Test for a Point D*

A specific property of the Delaunay triangulation is that no point will lie within the *circumcircle*[6] prescribed by the vertices of any other triangle. This is defined by the Incircle property, Figure 3.4, above.

---

[5] Circumcentre: The centre of a circle prescribed by three points lying on its perimeter.

[6] Circumcircle: The unique circle connecting the three points of a given triangle

Before triangulation takes place, an initial set of nodes defining the perimeter of the flow domain is required, and on this an initial triangulation is generated. There are four methods in which the Delaunay triangulation is generated:

- Incremental Insertion Algorithms; Guibas & Stolfi (1982)
- Divide and Conquer Algorithm; Dwyer (1987)
- Advanced Front Algorithm; Guibas & Stolfi (1985)
- Edge Swapping Algorithm; Lawson (1977)

3.4.1   Incremental Insertion Algorithms

An initial trianglulation is defined that encompasses all the points of the data set. The points that define the boundaries are inserted into the initial domain followed by a series of points defining a farfield boundary if that is required. A triangulation is now defined; to complete the process new points are continually added to the point set at the *circumcentre* of any triangle that exceeds some predefined criterion. This may be that the minimal internal angle of the triangle is greater than some limit or the ratio of the radius of the *circumcircle* to the radius of the in-circle is less than some limit.

(a) Before Insertion   **(b)** Delaunay Cavity   **(c)** New triangulation
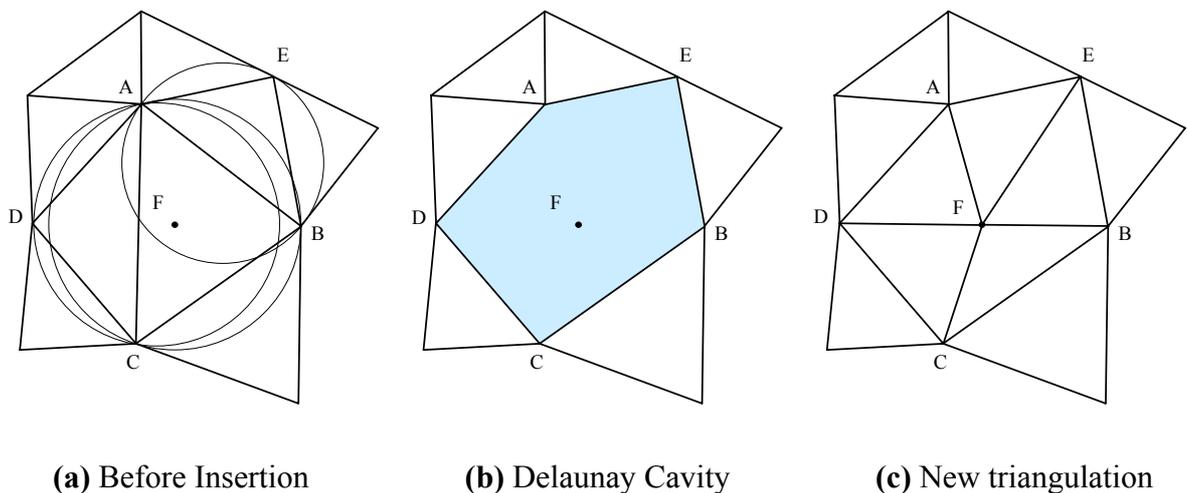
Figure 3.5: *Insertion of a New Point into ΔABC*

All existing triangles that contain the new point within their *circumcircle* are removed from the triangulation to leave a cavity and a new triangulation is made by connecting all nodes on the boundary of this cavity to the most recently inserted point. This can be seen in Figure 3.5,

for the insertion of point F into triangle ABC. This compromises triangles ACD and ABE requiring their removal, and points A through to E are then connected to the new point F. This process repeats until the Delaunay properties are valid for all points and triangles. This is also known as the Bowyer-Watson algorithm: Bowyer (1981), Watson (1981).

3.4.2   Divide and Conquer Algorithm

For the Divide and Conquer algorithm, the set of points must already be specified. The process is recursive in that the set of points is continually subdivided into two halves. Each half is Delaunay triangulated and the two halves are then merged together. The points must all be sorted in the x-axis. The main difficulty with this method is the merging of the two halves. There are two properties for the merging process:

1. Only cross edges are created in the process, i.e. only edges linking the halves.
2. Vertices with maximum (minimum) y values always connect.

The algorithm begins by forming a cross edge between the maximal y value points of each set, satisfying property 2. Let this edge be denoted AB. A circle of increasing radius is then placed on AB such that A and B lie on the circle. The circle will then increase upwards and away from the line as can be seen in Figure 3.6:
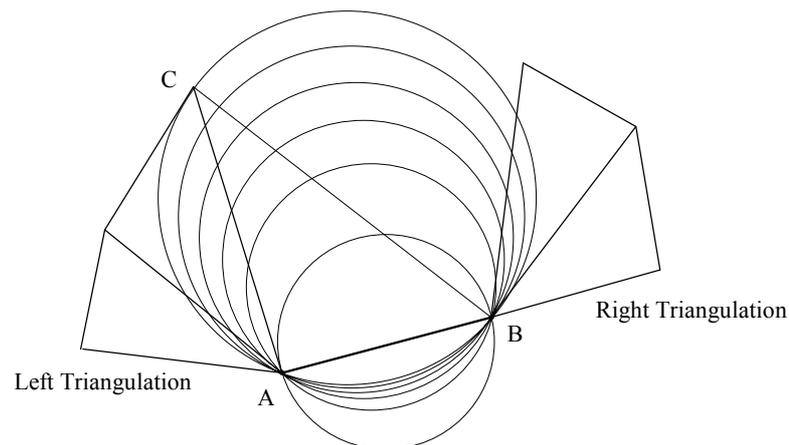
Figure 3.6: *Generating Next Cross Edge*

The circle will eventually encounter a point belonging to either of the two sets. A new cross-edge is added that connects the new point with the end of line AB in the opposite set. In the

above case, C belongs to the left set, hence the new cross-edge connects C and point B which is in the right set. Edge CB now becomes the active edge and the process repeats until both the maximum y values have been reached. Any edges that are intersected are then removed from the triangulation. These will be edges from the left to the left set, and the equivalent in the right set:

Edges marked
for removal

(a) Defining Cross-Edges   **(b)** Merged Triangulations
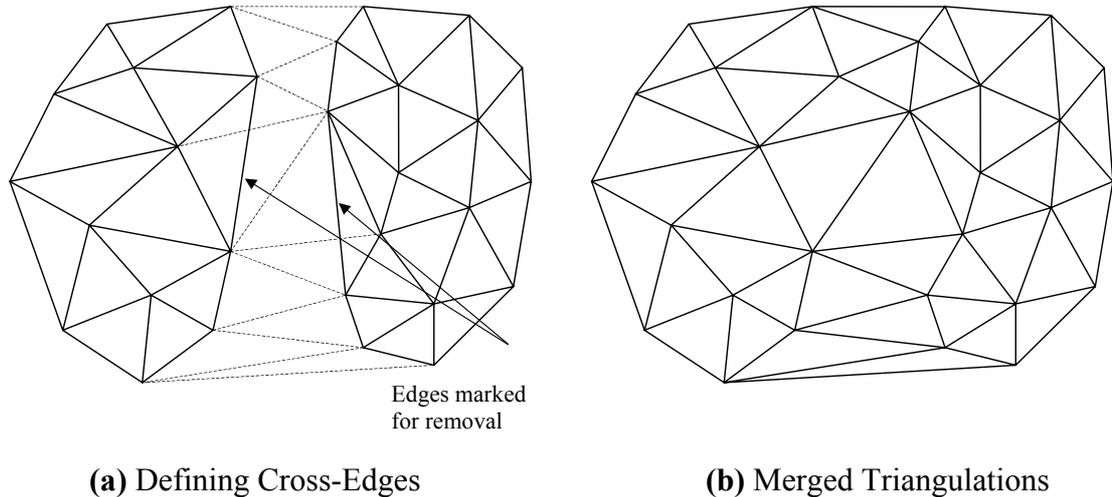
Figure 3.7: *Merging Left and Right Triangulations*

Note the two intersections on the left image in Figure 3.7. It should be noted that at every level of recursion through the algorithm, all data sets are convex.

3.4.3   Advanced Front Algorithm

Unlike the Divide and Conquer routine, the Advanced Front Algorithm, Tanemura *et al* (1983), Merriam (1991), does not sub-divide the data set. The process begins by defining all the boundary edges. All these edges are available to the Advanced Front Routine. This will be called the Available Edges List. One now starts with an edge belonging to the available edges list. A Triangle is generated by joining the ends of this edge to one of the interior points.
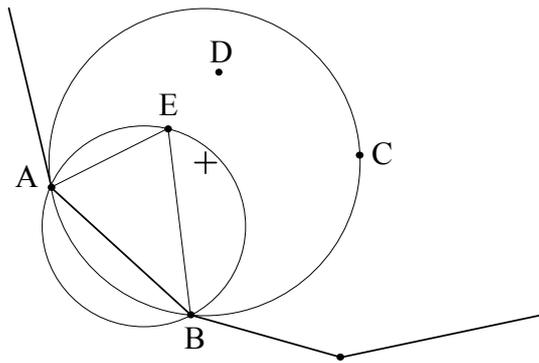
Figure 3.8: *Advanced Front Algorithm*

The algorithm proceeds by selecting any interior point (point C for example in Figure 3.8), a circumcircle prescribed by the end points and the new point is constructed. If there are any other interior points within this circumcircle (such as D and E) then replace the selected point with the point that lies nearest to the centre of the cirumcircle. From above, point E is closest. When the circumcircle is empty, connect the endpoints to the selected point:

These new edges any now added to the Available Edges List and the original edge AB is removed. The process now iterates until there are no more edges in the list.

3.4.4   Edge Swapping Algorithm

The edge swapping method begins with a simple triangulation of the point set that does not satisfy the Delaunay constraints. The set is then made to conform to the Delaunay criterion by application of the edge swapping algorithm such that the *equiangularity* A(T) of the triangulation increases. A(T) is defined as the ordering of the angles A(T) = [α_1, α_2, α_3, ... α_{3n}] such that $\alpha_i \leq \alpha_j$ if i < j.

A triangulation A(T*) < A(T) if $\alpha_j^* \leq \alpha_j$ and $\alpha_i^* = \alpha_i$ for $1 \leq i < j$.

In the algorithm, all interior edges of the triangulation are examined. Each is taken to be the diagonal of the quadrilateral formed from the two triangles sharing the edge. The quadrilateral is checked to make sure that a potential edge swap does not cause an intersection of the opposite edges.

If it is convex, then the diagonal is chosen that maximises the *equiangularity* of the quadrilateral. This is equivalent to maximising the minimum angle of the two adjacent triangles, Figure 3.9.

Current Diagonal α < β   Diagonal Swapped
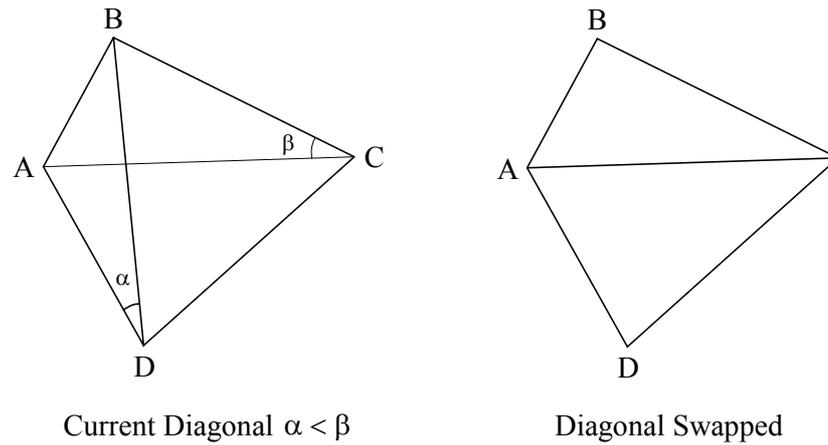
Figure 3.9: *Choosing the Diagonal*

If a mesh is locally equiangular everywhere then it can be shown to be a Delaunay triangulation.

3.5   Concluding Remarks

Various methods of triangulation have been presented for the generation of unstructured meshes. For the current study the triangulation software 'Triangle' be Jonathon Shewchuk has been used. The software is of a very high quality and generates satisfactory meshes. This has enabled the author to pursue other areas of the research. The particular method used is the *divide and conquer* algorithm.

4   GOVERNING EQUATIONS

In this chapter a numerical method is described to solve the unsteady, compressible Euler equations.

Roe's *Upwind* Flux Difference Splitting method is employed together with a higher order scheme; Roe (1981). This is implemented on an unstructured grid constructed from a Delaunay triangulation with the conserved variables stored at the vertices of the mesh. Solutions are generated using fully implicit, steady and unsteady time integration.

The governing equations are derived from the mathematical equations defining conservation of mass, conservation of momentum and conservation of total energy for a viscous fluid:

- Conservation of mass:
$$\frac{\partial}{\partial t}\rho + \nabla \cdot (\rho \boldsymbol{U}) = 0 \qquad (4.1)$$

- Conservation of momentum:
$$\frac{\partial}{\partial t}\rho \boldsymbol{U} + \nabla \cdot (\rho \boldsymbol{U} \otimes \boldsymbol{U} + p\boldsymbol{I} - \boldsymbol{\tau}) = 0 \qquad (4.2)$$

- Conservation of total energy:
$$\frac{\partial}{\partial t}\rho e_0 + \nabla \cdot (\rho \boldsymbol{U} h_0 - \boldsymbol{\tau} \cdot \boldsymbol{U} - \dot{\boldsymbol{q}}) = 0 \qquad (4.3)$$

where ρ is the density, $\boldsymbol{U}$ is the velocity vector, p the pressure, $\boldsymbol{I}$ the identity matrix, $\boldsymbol{\tau}$ the shear stress tensor, e_0 the total energy per unit volume, h_0 the total enthalpy per unit volume, and $\dot{\boldsymbol{q}}$ the rate of heat transfer vector. $\otimes$ is the dyadic product. This is an *Eulerian*[7] principle where the properties are expressed as a function of space and time, as opposed to *Lagrangian*[8] where the properties are considered by following a particle.

The following sections describe the major components in the development of the unstructured flow solver.

---

[7] Eulerian: Leonard Euler (1707-1783). Swiss mathematician who first worked out the equations of motion for fluids. He was the leading mathematician of his time.

[8] Lagrangian: Joseph Louis Lagrange (1736-1813). French Mathematician.

4.1   The Euler Equations

The Euler Equations are derived from the Navier-Stokes equations by considering the flow to be inviscid[9], adiabatic and a continuum[10]. For very large Reynolds number flow $Re \rightarrow \infty$, the ratio $1/Re \rightarrow 0$ and so the Euler equations become a fair approximation to physical flow conditions in the limits of vanishing viscosity. The Euler equations can be expressed in various forms depending on the variables under consideration. The three forms most commonly used are: Conservative form; the variables used are the density, the x and y momentum components and the total energy, Primitive form; the variables are the density, x and y velocity components and the static pressure, and the Characteristic form where the dependent variables are those variables that propagate along the characteristic lines. From a theoretical point of view, it can be shown that the actual form to be used is immaterial for subsonic or supersonic flow; simple transformations can be used to change between methods. However, from a numerical point of view, there is the problem that discontinuities can be present in the flow field in the form of shock waves. Under these circumstances, the primitive (differential) form of the Euler equations is meaningless, as their derivatives do not exist. In this situation only the conservative form of the Euler equations can be used.

4.2   The Navier-Stokes Equations

The 'quasi-3D'[11] Reynolds Averaged Navier-Stokes equations define a system of non-linear, parabolic/elliptic, 2nd order conservation laws describing the flow of a compressible, viscous, Newtonian[12] fluid, neglecting body forces[13] and chemical reactions[14]. The set of equations is derived from the application of Newton's 2nd law of motion to a unit body of fluid. These laws represent the conservation of mass, momentum in the x and y directions (as the model is assumed to be two-dimensional), and total energy. The Euler equations can be seen as the

---

[9] Inviscid: Non-viscous; without viscosity, the fluid contains no shear stresses.

[10] Continuum: A fluid in which the distance between fluid particles is smaller than the mean free path, defined as the distance a particle has to move before colliding with another.

[11] Quasi-3D: 3-dimensional flow is simulated by the addition of a streamtube thickness defined at each point.

[12] Newtonian Fluid: A fluid in which the shear stresses are proportional to the deformation.

[13] Body Forces: are defined as gravitational, electromagnetic and buoyancy effects.

[14] Chemical Reactions: the flow is treated is being inert.

approximation of convection dominated flow at high Reynolds numbers, whereas the Navier-Stokes equations define flow that is diffusion dominated. They are therefore most applicable where the flow encountered has a low Reynolds number. Due to the effects of viscosity and heat conduction, inviscid discontinuities do not theoretically exist in this flow, being transformed into a continuous but sharp gradient across the discontinuity. In real flow, the shock wave has a discrete thickness that can be resolved by a viscous calculation based on the Mach number *M*. The thickness is defined by the balance between the heat conduction and viscous effects opposing the pressure, and inertia forces across the shock wave. For a Mach number of 1.5 this gives a shock thickness of 10^{-4}mm. As such, this is still generally thinner that can be accurately meshed so the flow can still be treated as containing discontinuities.

4.3   Conservative form of the Euler Equations

The conservative flux vector form of the Euler equations is defined as:

$$\frac{\partial (\boldsymbol{Q})}{\partial t} + div(\boldsymbol{E}(\boldsymbol{Q})) = 0 \tag{4.4}$$

where the conserved variables $\boldsymbol{Q}$, and the inviscid Euler flux vector $\boldsymbol{E}(\boldsymbol{Q})$ are defined as:

$$\boldsymbol{Q} = \left\{ \begin{array}{c} \rho \\ \rho u \\ \rho v \\ \rho e_0 \end{array} \right\} \qquad \boldsymbol{E}(\boldsymbol{Q}) = \left\{ \begin{array}{c} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(\rho e_0 + p) \end{array} \right\} i + \left\{ \begin{array}{c} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(\rho e_0 + p) \end{array} \right\} j \tag{4.5}$$

Variables ρ, *u* and *v* are the density and the velocity components in the *x* and *y* directions, e_0 is the total energy per unit volume, and *p* is the static pressure.

The equations are closed with the pressure equation:

$$p = (\gamma - 1)\left\{ \rho e_0 - \frac{\rho(u^2 + v^2)}{2} \right\} \tag{4.6}$$

where γ is the ratio of specific heats $\dfrac{C_P}{C_V}$ and is generally taken as 1.4 for air.

4.4   Jacobian Matrices with respect to the Conservative Variables

The Jacobian matrices are partial derivates of $E(Q)$ with respect to the conservative variables Q and are derived as follows:

Q is first expressed as the variables Q_i, i = 0 to 3 to give:

$$Q = \begin{Bmatrix} Q_0 \\ Q_1 \\ Q_2 \\ Q_3 \end{Bmatrix} \tag{4.7}$$

Expressing $E(Q)$ in terms of Q as above gives:

$$E(Q) = \begin{Bmatrix} Q_1 \\ \dfrac{Q_1^2}{Q_0} + (\gamma-1)\left(Q_3 - \dfrac{(Q_1^2+Q_2^2)}{2Q_0}\right) \\ \dfrac{Q_1 Q_2}{Q_0} \\ \gamma\dfrac{Q_1 Q_3}{Q_0} - (\gamma-1)\left(\dfrac{Q_1(Q_1^2-Q_2^2)}{2Q_0^2}\right) \end{Bmatrix} i + \begin{Bmatrix} Q_2 \\ \dfrac{Q_1 Q_2}{Q_0} \\ \dfrac{Q_2^2}{Q_0} + (\gamma-1)\left(Q_3 - \dfrac{(Q_1^2+Q_2^2)}{2Q_0}\right) \\ \gamma\dfrac{Q_2 Q_3}{Q_0} - (\gamma-1)\left(\dfrac{Q_2(Q_1^2-Q_2^2)}{2Q_0^2}\right) \end{Bmatrix} j \tag{4.8}$$

The Jacobian matrices in the *x* and *y* directions $\dfrac{\partial E_x(Q)}{\partial Q}$ and $\dfrac{\partial E_y(Q)}{\partial Q}$ are:

$$\frac{\partial E_x(Q)}{\partial Q} = \begin{Bmatrix} 0 & 1 & 0 & 0 \\ \dfrac{(\gamma-3)Q_1^2 + (\gamma-1)Q_2^2}{2Q_0^2} & -\dfrac{(\gamma-3)Q_1}{Q_0} & -\dfrac{(\gamma-1)Q_2}{Q_0} & (\gamma-1) \\ -\dfrac{Q_1 Q_2}{Q_0^2} & \dfrac{Q_2}{Q_0} & \dfrac{Q_1}{Q_0} & 0 \\ -\dfrac{\gamma Q_1 Q_3}{Q_0^2} + \dfrac{(\gamma-1)Q_1(Q_1^2+Q_2^2)}{Q_0^3} & \dfrac{\gamma Q_3}{Q_0} - \dfrac{(\gamma-1)(3Q_1^2+Q_2^2)}{2Q_0^2} & -\dfrac{(\gamma-1)Q_1 Q_2}{Q_0^2} & \dfrac{\gamma Q_1}{Q_0} \end{Bmatrix}$$

$$\tag{4.9}$$

and

$$\frac{\partial E_y(Q)}{\partial Q} = \begin{Bmatrix} 0 & 0 & 1 & 0 \\ -\dfrac{Q_1 Q_2}{Q_0^2} & \dfrac{Q_2}{Q_0} & \dfrac{Q_1}{Q_0} & 0 \\ \dfrac{(\gamma-1)Q_1^2 + (\gamma-3)Q_2^2}{2Q_0^2} & -\dfrac{(\gamma-1)Q_1}{Q_0} & -\dfrac{(\gamma-3)Q_2}{Q_0} & (\gamma-1) \\ -\dfrac{\gamma Q_2 Q_3}{Q_0^2} + \dfrac{(\gamma-1)Q_2\left(Q_1^2 + Q_2^2\right)}{Q_0^3} & -\dfrac{(\gamma-1)Q_1 Q_2}{Q_0^2} & \dfrac{\gamma Q_3}{Q_0} - \dfrac{(\gamma-1)\left(Q_1^2 + 3Q_2^2\right)}{2Q_0^2} & \dfrac{\gamma Q_2}{Q_0} \end{Bmatrix}$$

$$(4.10)$$

The Jacobian matrices are necessary for the spatial discretisation based on a Taylor series expansion described in the following text.

4.5   Spatial Discretisation

For the spatial discretisation, a finite volume method is employed based on a cell-vertex scheme, Barth & Jesperson (1989).

The main steps to the method are:

- The flow domain under investigation is first subdivided into an unstructured triangular mesh Ω by an appropriate method; in this case a Delaunay triangulation.
- Node points are defined at the vertices of the mesh.
- A control volume Ω_n is associated with each of these nodes and is constructed by taking the triangles sharing a common vertex *n*, joining the centroids of adjacent triangles and the mid-points of the edges to form a boundary; the control volume is defined as the area within this boundary.

Figure 4.1 shows the control volume associated with node *n* and the adjacent control volume associated with the neighbouring node *k*. When applied to all the nodes, this generates a set of non-overlapping cells covering the whole domain. This is known as the *Median Dual* of the triangulation. In this scheme the entries for Q are stored at each of the nodes in the mesh, and are assumed to be the cell-averaged quantities of Q.
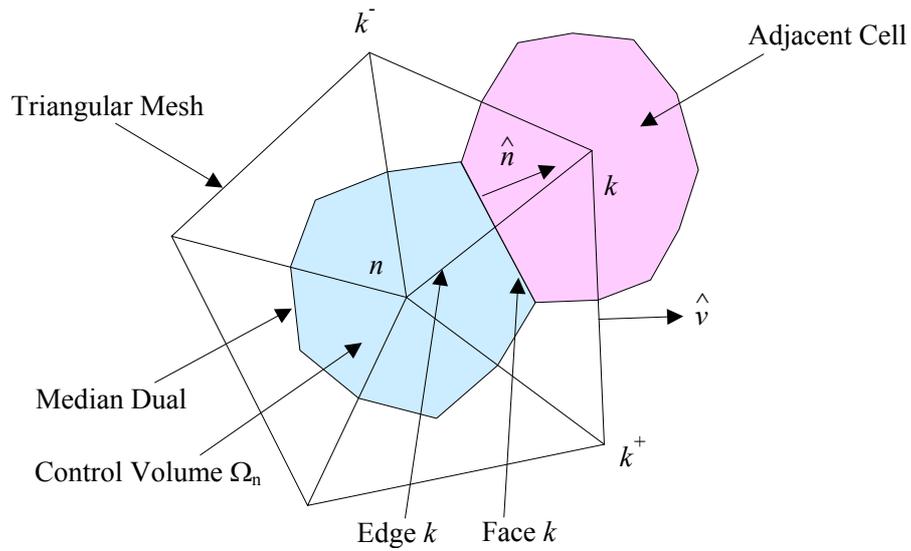
Figure 4.1: *Median Dual Control Volume*

The conservative form of the (4.4) can be expressed in its integral form:

$$\frac{\partial}{\partial t}\iint_{\Omega_n} \boldsymbol{Q}\,dA = -\iint_{\Omega_n} \nabla \cdot \boldsymbol{E}(\boldsymbol{Q})\,dA \tag{4.11}$$

By applying Gauss' divergence theorem, each of the area integrals can be expressed as a closed-loop line integral around the boundary of the cell. Applying to (4.11) gives:

$$\frac{\partial}{\partial t}\iint_{\Omega_n} \boldsymbol{Q}\,dA = -\oint_{\Omega_n} \boldsymbol{E}(\boldsymbol{Q}) \cdot \hat{n}\,dl \tag{4.12}$$

For the left hand side, the rate of change of $\boldsymbol{Q}$ is assumed to be constant over the cell, therefore it can be discretised as:

$$\frac{\partial}{\partial t}\iint_{\Omega_n} \boldsymbol{Q}\,dA = \frac{\Delta \boldsymbol{Q}}{\Delta t} A_n \tag{4.13}$$

where $\Delta \boldsymbol{Q}$ is the change in $\boldsymbol{Q}$ between consecutive two time steps and Δt is the time step.

The boundary of the control volume is seen to consist of a finite number of short edges, Figure 4.2, the line integral will consist of these edges.
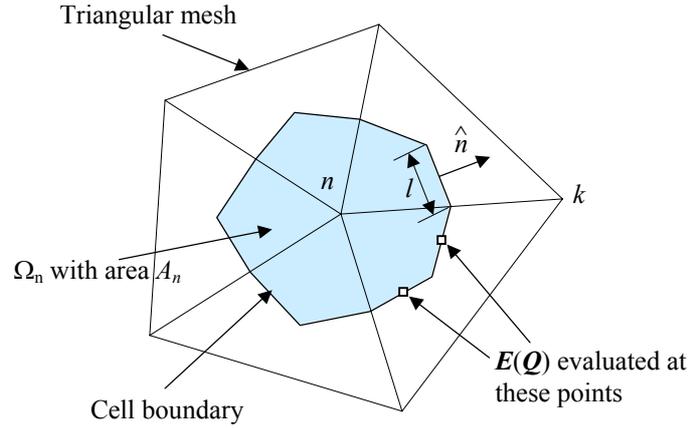
Figure 4.2: *Line Integral Calculation*

If the property to be integrated is considered to be constant along each edge, and equal to the value at the centre of that edge, the line integral can be expressed as a summation of these values multiplied by the corresponding normal vector to the edge. The discretised form then becomes:

$$\frac{\Delta \boldsymbol{Q}}{\Delta t} = -\frac{1}{A_n} \sum_{k \in K(n)} \boldsymbol{E}(\boldsymbol{Q}_{n,k}) \cdot \hat{n}l \tag{4.14}$$

where $K(n)$ is the set of adjacent nodes to the central node n, $\boldsymbol{E}(\boldsymbol{Q}_{n,k})$ denotes the property $\boldsymbol{E}(\boldsymbol{Q})$ evaluated at the boundary between adjacent nodes n and k, $\hat{n}$ is the outward unit normal vector of each segment, with l being the length of that segment, and A_n is the area of cell Ω_n.

4.6   Upwinding Schemes

The flow solver in this research uses a technique in which the space and time discretisation are treated independently. Both types will be discussed in the following sections. There are two types of spatial discretisation: *central differencing* and *upwinding* schemes.

The term *upwinding* is derived from the manner in which the fluxes are evaluated according to the direction of wave propagation as opposed to *central differencing* schemes in which the fluxes are evaluated via some numerical averaging technique applied to the left and right states. Central differencing schemes are therefore highly insensitive to the propagation of properties within the flow. The effect of this is dependent on the conditions of the flow under investigation; this means that a flow solver of this type would lose generality. A particular

phenomenon called 'chequer-boarding' presents itself in such solvers in which alternate cells in a discretisation can hold alternating values, in much the same way as a chess board contains alternating black and white squares. This could be mis-interpreted as a very high frequency oscillation present in the flow; it should be noted that these oscillations are entirely spurious and not a property of the flow, Morton (1996). It is therefore necessary to employ some form of artificial dissipation into the scheme, typically in the form of fourth order damping terms to remove these oscillations. The actual level of the damping that is required is a question of trial and error, relating to the flow conditions being modelled, and so stability and accuracy can be hard to achieve. Finally and more importantly, central differencing schemes do not take into account any of the aforementioned directional properties of a hyperbolic system, a property inherent to the Euler equations. It is these disadvantages that have led to the development of *upwinding* schemes.

In the *upwinding* scheme, the flux at the interface of the cell is defined by the left and right states $\boldsymbol{Q}_n$ and $\boldsymbol{Q}_k$. The flux function takes into account the relevant propagating waves by the sign of the eigenvalues of the Jacobian matrices $\dfrac{\partial E(\boldsymbol{Q}_n)}{\partial \boldsymbol{Q}}$ and $\dfrac{\partial E(\boldsymbol{Q}_k)}{\partial \boldsymbol{Q}}$. In order to be consistent with the hyperbolic nature of the Euler equations, the upwinding scheme must satisfy the following properties:

1.   For all positive eigenvalues: $\boldsymbol{E}(\boldsymbol{Q}_{n,k}) = \boldsymbol{E}(\boldsymbol{Q}_n)$

       For all negative eigenvalues: $\boldsymbol{E}(\boldsymbol{Q}_{n,k}) = \boldsymbol{E}(\boldsymbol{Q}_k)$

       i.e. for supersonic flow from the left, the cell interface is only affected by the left (upstream) state. The same applies for supersonic flow from the right.

2.   For a combination of positive and negative eigenvalues (subsonic flow) a Taylor series expansion about a point $\boldsymbol{Q}^*$ bounded by $\boldsymbol{Q}_n$ and $\boldsymbol{Q}_k$ is used to defined an intermediate state:

$$\boldsymbol{E}(\boldsymbol{Q}_{n,k}) = \boldsymbol{E}(\boldsymbol{Q}^*) + \frac{\partial \boldsymbol{E}^+(\boldsymbol{Q}^*)}{\partial \boldsymbol{Q}}\left(\boldsymbol{Q}_n - \boldsymbol{Q}^*\right) + \frac{\partial \boldsymbol{E}^-(\boldsymbol{Q}^*)}{\partial \boldsymbol{Q}}\left(\boldsymbol{Q}_k - \boldsymbol{Q}^*\right) \qquad \textbf{(4.15)}$$

    where $\dfrac{\partial \boldsymbol{E}^+(\boldsymbol{Q}^*)}{\partial \boldsymbol{Q}} = T\Lambda^+ T^{-1}$ containing positive eigenvalues and $\dfrac{\partial \boldsymbol{E}^-(\boldsymbol{Q}^*)}{\partial \boldsymbol{Q}} = T\Lambda^- T^{-1}$

    containing negative eigenvalues.

3.   For equal states there is no flux: $\boldsymbol{E}(\boldsymbol{Q}_{n,n}) = 0$

Thus the properties of the Euler equations are preserved.

There are two major families of *upwinding* schemes for calculating $E(Q_{n,k})$: *flux differencing* and *flux vector splitting*. Flux differencing schemes are based on the principle of the Riemann Solver whereas Flux vector splitting schemes are based on analysing the left and right states and then combining the results.

4.6.1   Flux Differencing Schemes: Riemann Solvers

Riemann solvers generate a solution by considering each cell interface as a *Riemann Problem* (also called a shock-tube problem), Hirsch (1990). It is of particular interest, as it presents an exact solution to the one-dimensional Euler equations. This is a discontinuity between two states, each with its own wave properties. The flux across the face is then a result of the interaction between these opposing wave properties. This can be seen in Figure 4.3:
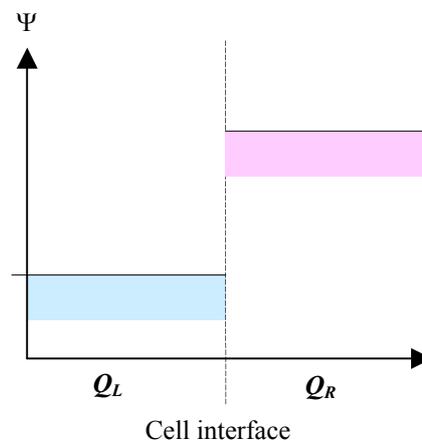
Figure 4.3: *Riemann Problem*

Q_L is the state in the control volume associated with the node on the left-hand side of the interface, Q_R the state on the right-hand side and Ψ is a property of Q. There are two approaches to solving this problem: *Exact Riemann Solvers* and *Approximate Riemann Solvers*. A detailed study of Riemann solvers can be found in Toro (1999).

4.6.1.1 Exact Riemann Solver

In a true Godunov scheme, all the Riemann Problems are solved exactly, and the results are combined to generate a solution for the whole domain, Godunov (1959). This can be achieved if the time step is sufficiently small so that there is no interference from neighbouring wave fronts. As the solution is exact, the Godunov scheme is monotonic; generating no new extrema, and satisfies the entropy condition. However, since the local function involves the same non-linear wave equations as the global function, the non-linearity implies that the function at the interface must be iterated to a solution. This is time-consuming and undesirable. As an alternative to this, several approximate solvers have been developed, the most notable of which was by Roe (1981). These offer a more manageable way of calculating the flux difference without resorting to iteration, whilst still retaining the properties of the hyperbolic equations. Roe's method is very popular due to its robustness and adaptability to upwinding and higher order schemes.

4.6.1.2 Roe's Approximate Riemann Solver

For Roe's Approximate Riemann Solver, Roe (1981), the flux at the interface is defined as follows:

$$\boldsymbol{E}(\boldsymbol{Q}_{n,k}) \cdot \hat{n} = \frac{1}{2} \left\{ \left(\boldsymbol{E}(\boldsymbol{Q}_n) + \boldsymbol{E}(\boldsymbol{Q}_k) \right) \cdot \hat{n} - \left| \widetilde{A} \right| (\boldsymbol{Q}_k - \boldsymbol{Q}_n) \right\} \tag{4.16}$$

The following method is used to calculate the term $\left| \widetilde{A} \right| (\boldsymbol{Q}_k - \boldsymbol{Q}_n)$, which defines the dissipation between the adjacent flow states. $\left| \widetilde{A} \right|$ is known is the Roe matrix and is a function of these two flow states. It is first approached by considering the Euler equation:

$$\frac{\partial \boldsymbol{Q}}{\partial t} + div(\boldsymbol{E}(\boldsymbol{Q})) = 0 \tag{4.17}$$

this is linearised to give

$$\frac{\partial \boldsymbol{Q}}{\partial t} + A \, div(\boldsymbol{Q}) = 0 \tag{4.18}$$

where the Jacobian matrix $A = \dfrac{\partial E(\boldsymbol{Q})}{\partial \boldsymbol{Q}}$. Replacing A with the required approximation $\tilde{A}$ gives:

$$\frac{\partial \boldsymbol{Q}}{\partial t} + \tilde{A}\,div(\boldsymbol{Q}) = 0 \tag{4.19}$$

$\tilde{A}$ is now computed by evaluating $\dfrac{\partial E(\boldsymbol{Q})}{\partial \boldsymbol{Q}}$ with a set of Roe Averaged Quantities. The approximation must satisfy 3 constraints set down by Roe for it to be consistent with the exact solution, namely:

- $\tilde{A}$ has real eigenvalues with linearly independent eigenvectors, this ensures that the local system stays hyperbolic.

- Given any left and right states $\boldsymbol{Q}_n$ and $\boldsymbol{Q}_k$ there exists a unique solution:

$$E(\boldsymbol{Q}_k) - E(\boldsymbol{Q}_n) = \tilde{A}\big[\boldsymbol{Q}_k - \boldsymbol{Q}_n\big] \tag{4.20}$$

ensuring homogeneity of the equations.

- Given two states $\boldsymbol{Q}_n = \boldsymbol{Q}_k$, then $\tilde{A} \equiv \dfrac{\partial E(\boldsymbol{Q})}{\partial \boldsymbol{Q}}$, i.e. the approximate solution is equivalent to the exact solution if the left and right states are equal.

The above properties are satisfied if the Roe matrix is evaluated by means of square root density-averaged variables defined as follows:

$$\omega = \sqrt{\frac{\rho_R}{\rho_L}}$$

$$\tilde{\rho} = \frac{(\rho_L + \rho_R \omega)}{(1+\omega)} \qquad \tilde{u} = \frac{(u_L + u_R \omega)}{(1+\omega)} \qquad \tilde{v} = \frac{(v_L + v_R \omega)}{(1+\omega)} \qquad \tilde{p} = \frac{(p_L + p_R \omega)}{(1+\omega)} \tag{4.21}$$

Mach velocity, enthalpy and flow property gradients can then be defined as:

$$\tilde{a} = \sqrt{\frac{\gamma \tilde{p}}{\tilde{\rho}}} \quad \text{and} \quad \tilde{h} = \frac{\tilde{a}^2}{(\gamma - 1)} + \frac{\tilde{u}^2 + \tilde{v}^2}{2} \tag{4.22}$$

$$\Delta u = u_R - u_L$$
$$\Delta v = v_R - v_L$$
$$\Delta p = p_R - p_L$$
$$\Delta \rho = \rho_R - \rho_L$$
$$\widetilde{U} = \widetilde{u}\hat{n}_x + \widetilde{v}\hat{n}_y$$
$$\widetilde{V} = \widetilde{v}\hat{n}_x - \widetilde{u}\hat{n}_y$$
$$\Delta U = \hat{n}_x \Delta u + \hat{n}_y \Delta v$$
$$\Delta V = \hat{n}_x \Delta v - \hat{n}_y \Delta u$$

(4.23)

where $\hat{n}$ is the unit vector of Median Dual face.

The matrix $\widetilde{A}$ can be decomposed by introducing diagonalising matrices $\widetilde{T}$ and $\widetilde{T}^{-1}$, to give:

$$\widetilde{A} = \widetilde{T}\widetilde{\Lambda}\widetilde{T}^{-1}$$

(4.24)

where $\widetilde{\Lambda}$ is the diagonal matrix containing the eigenvalues of $\widetilde{A}$.

$\left|\widetilde{A}\right|$ can then be defined as:

$$\left|\widetilde{A}\right| = \widetilde{T}\left|\widetilde{\Lambda}\right|\widetilde{T}^{-1}$$

(4.25)

giving

$$\left|\widetilde{A}\right|(\boldsymbol{Q}_k - \boldsymbol{Q}_n) = \widetilde{T}\left|\widetilde{\Lambda}\right|\widetilde{T}^{-1}\Delta\boldsymbol{Q}$$

(4.26)

The matrices $\left|\widetilde{A}\right|$, $\widetilde{T}$ and $\widetilde{T}^{-1}$ are defined as:

$$\left|\widetilde{\Lambda}\right| = \left\{\begin{matrix} \left|\widetilde{U}-\widetilde{a}\right| & 0 & 0 & 0 \\ 0 & \left|\widetilde{U}\right| & 0 & 0 \\ 0 & 0 & \left|\widetilde{U}\right| & 0 \\ 0 & 0 & 0 & \left|\widetilde{U}+\widetilde{a}\right| \end{matrix}\right\}$$

(4.27)

and

$$\widetilde{T} = \begin{Bmatrix} 1 & 1 & 0 & 1 \\ \widetilde{u} - \widetilde{a}\hat{n}_x & \widetilde{u} & -\hat{n}_y & \widetilde{u} + \widetilde{a}\hat{n}_x \\ \widetilde{v} - \widetilde{a}\hat{n}_y & \widetilde{v} & \hat{n}_x & \widetilde{v} + \widetilde{a}\hat{n}_y \\ \widetilde{h} - \widetilde{a}\widetilde{U} & \frac{1}{2}\left(\widetilde{u}^2 + \widetilde{v}^2\right) & \widetilde{V} & \widetilde{h} + \widetilde{a}\widetilde{U} \end{Bmatrix}$$

(4.28)

$$\widetilde{T}^{-1} = \begin{Bmatrix} \dfrac{\widetilde{U}}{2\widetilde{a}} + \dfrac{\zeta_0}{2} & 1 - \zeta_0 & -\widetilde{V} & -\dfrac{\widetilde{U}}{2\widetilde{a}} + \dfrac{\zeta_0}{2} \\ -\dfrac{\hat{n}_x}{2\widetilde{a}} - \dfrac{\zeta_1}{2} & \zeta_1 & -\hat{n}_y & \dfrac{\hat{n}_x}{2\widetilde{a}} - \dfrac{\zeta_1}{2} \\ -\dfrac{\hat{n}_y}{2\widetilde{a}} - \dfrac{\zeta_2}{2} & \zeta_2 & \hat{n}_x & \dfrac{\hat{n}_y}{2\widetilde{a}} - \dfrac{\zeta_2}{2} \\ \dfrac{\zeta_3}{2} & -\zeta_3 & 0 & \dfrac{\zeta_3}{2} \end{Bmatrix}$$

(4.29)

where:

$$\zeta_0 = \frac{(\gamma - 1)(\widetilde{u}^2 + \widetilde{v}^2)}{2\widetilde{a}^2}$$

$$\zeta_1 = \frac{\widetilde{u}(\gamma - 1)}{\widetilde{a}^2}$$

$$\zeta_2 = \frac{\widetilde{v}(\gamma - 1)}{\widetilde{a}^2}$$

$$\zeta_3 = \frac{(\gamma - 1)}{\widetilde{a}^2}$$

(4.30)

The four eigenvalues associated with the three wave speeds are:

$$\left|\widetilde{\lambda}_1\right| = \left|\widetilde{U} - \widetilde{a}\right|$$

$$\left|\widetilde{\lambda}_{2,3}\right| = \left|\widetilde{U}\right|$$

$$\left|\widetilde{\lambda}_4\right| = \left|\widetilde{U} + \widetilde{a}\right|$$

(4.31)

and their respective characteristic wave speeds are:

$$\widetilde{T}^{-1}\Delta\boldsymbol{Q} = \begin{Bmatrix} \dfrac{\Delta p - \widetilde{\rho}\widetilde{a}\Delta U}{2\widetilde{a}^2} \\ \Delta\rho - \dfrac{\Delta p}{\widetilde{a}^2} \\ \rho\Delta V \\ \dfrac{\Delta p + \widetilde{\rho}\widetilde{a}\Delta U}{2\widetilde{a}^2} \end{Bmatrix}$$

(4.32)

These correspond to:

- A backward moving *acoustic* wave

- An *entropy* wave

- A s*hear* wave

- A forward moving *acoustic* wave

4.6.1.3   Entropy Satisfying Condition

There is an inherent problem associated with Roe's Approximate Riemann Solver, that is its inability to correctly resolve entropy conditions over expansion shocks that occur at sonic points. At a sonic point one of the eigenvalues will be zero, so during the calculation its corresponding eigenvector will by multiplied by zero, and as a consequence there will be no dissipation associated with the characteristic wave. As a result, the expansion shock will be propagated, which is a physically impossible condition. However, the solution to this requires a simple modification to the eigenvalues.

An entropy-satisfying condition is employed by modifying the calculated eigenvalues. This explicitly introduces an expansion fan into the flow at sonic conditions by smoothing the eigenvalues of the Roe matrix.

The eigenvalues associated with wave speeds $\left|\widetilde{U} - \widetilde{a}\right|$ and $\left|\widetilde{U} + \widetilde{a}\right|$ are modified as follows:

$$\left|\widetilde{\lambda_i}\right| = \begin{cases} \left|\widetilde{\lambda_i}\right| & \left|\widetilde{\lambda_i}\right| \geq \dfrac{\varepsilon_i}{2} \\[4mm] \dfrac{4\widetilde{\lambda_i}^2 + \varepsilon_i^2}{4\varepsilon_i} & \left|\widetilde{\lambda_i}\right| < \dfrac{\varepsilon_i}{2} \end{cases} \qquad\qquad \textbf{(4.33)}$$

The value ε_i in the above equation is defined as:

$$\begin{aligned} \varepsilon_1 &= MAX\big[0, \quad 4*\big(\left(U_R - a_R\right) - \left(U_L - a_L\right)\big)\big] \\ \varepsilon_4 &= MAX\big[0, \quad 4*\big(\left(U_R + a_R\right) - \left(U_L + a_L\right)\big)\big] \end{aligned} \qquad \textbf{(4.34)}$$

where U_L and U_R are the normal velocities of the left and right states in the direction of the normal vector $\hat{n}$, and a_L and a_R are the corresponding Mach velocities. A modification to this by Harten & Hyman (1983), generates a continuously differentiable solution with the value a ε_i defined as:

$$\varepsilon_i = MAX\left[0,\ \ \left(\widetilde{\lambda}_i - \lambda_{Li}\right),\ \ \left(\lambda_{Ri} - \widetilde{\lambda}_i\right)\right]$$

(4.35)

where $\widetilde{\lambda}_i$ is the ith Roe averaged eigenvalue and λ_{Li} and λ_{Ri} are the ith eigenvalues of the left and right states as calculated above. The new eigenvalues $\widetilde{\lambda}_i$ are then defined as:

$$\left|\widetilde{\lambda}_i\right| = \begin{cases} \left|\widetilde{\lambda}_i\right| & \left|\widetilde{\lambda}_i\right| \geq \varepsilon_i \\[2mm] \dfrac{1}{2}\left(\dfrac{\widetilde{\lambda}_i^{\,2}}{\varepsilon_i} + \varepsilon_i\right) & \left|\widetilde{\lambda}_i\right| < \varepsilon_i \end{cases}$$

(4.36)

A fundamental assumption of Roe's scheme is that the characteristic waves are propagated in the direction of the face normal vector $\hat{n}$. If the waves are indeed lined up with the face then Roe's scheme will exactly resolve the interface. For non-aligned waves, an error is introduced in which a proportion of the contact discontinuity is interpreted as a pressure wave. Rotated Riemann Solvers are extensions of Roe's scheme that deal with these issues. They are not discussed here, but have been presented in detail by Levy (1990) and Rumsey (1991).

4.6.2   Flux Vector Splitting

Flux vector splitting techniques split the interface into two components dependent on the left and right states. The direction of wave propagation is taken into account by splitting each of the Jacobian matrices into positive and negative components defined by the positive and negative eigenvalues. The cell interface will then be affected by the upstream components (positive eigenvalues) of the left face and the downstream components (negative eigenvalues) of the right face, that is, if the flow direction was from left to right. If the flow direction were from right to left then the opposite would be true.

The flux at the interface is dependent on the left and right states:

$$E(Q_{n,k}) = E^+(Q_n) + E^-(Q_k) \tag{4.38}$$

The notation, *n* and *k* for the left and right states is continued here, for consistency with the previous sections.

There are two main methods for flux vector splitting, those originally developed by Steger and Warming (1981) and the subsequent Van Leer scheme (1982).

4.6.2.1   Steger and Warming Flux Vector Splitting

In it simplest form, the left and right fluxes are calculated independently and then added to give the interface flux. As the Euler equations are hyperbolic, the Jacobian can be diagonalised with real eigenvalues:

$$E(Q) = \frac{\partial E(Q)}{\partial Q}.Q \qquad \text{where} \qquad \frac{\partial E(Q)}{\partial Q} = T\Lambda T^{-1} \tag{4.39}$$

The upstream and downstream Jacobians are defined by the positive and negative eigenvalues:

$$\frac{\partial E^+(Q)}{\partial Q} = T\Lambda^+ T^{-1} \qquad \text{and} \qquad \frac{\partial E^-(Q)}{\partial Q} = T\Lambda^- T^{-1} \tag{4.40}$$

where $\Lambda^+ = \dfrac{\Lambda + |\Lambda|}{2}$ and $\Lambda^- = \dfrac{\Lambda - |\Lambda|}{2}$. The flux at the interface can then be reconstructed as:

$$E(Q_{n,k}) = \frac{\partial E^+(Q_n)}{\partial Q}.Q_n + \frac{\partial E^-(Q_k)}{\partial Q}.Q_k \tag{4.41}$$

Unfortunately, the disadvantage of this scheme is that when an eigenvalue changes sign, a discontinuity is generated which has a non-unique solution. This would generate ambiguous errors affecting the stability of the whole scheme. To overcome this problem, Van Leer developed a scheme in which the solution at the interface is unique and differentiable, through a series of second and fourth order polynomials based on the Mach number of the interface.

4.6.2.2   Van Leer Flux Splitting

Under the Van Leer scheme, the flux at the interface is evaluated as:

$$E(Q_{n,k}) = E^+(Q_n) + E^-(Q_k) \tag{4.42}$$

The Jacobian matrices are split according to the Mach number at the cell interface.
For supersonic flow ($M_n \geq 1$) in the direction of the unit face normal $\hat{n}$:

$$E^+(Q_n) = E(Q_n) \qquad\qquad E^-(Q_k) = 0 \tag{4.43}$$

and for supersonic flow in the opposite direction ($M_n \leq -1$):

$$E^+(Q_n) = 0 \qquad\qquad E^-(Q_k) = E(Q_k) \tag{4.44}$$

For those cases where the Mach number is subsonic; $-1 < M_n < 1$, the flux is defined as:

$$E^\pm(Q_n) = \pm\frac{\rho a(1 \pm M_n)^2}{4}\left\{\begin{array}{c} u + \dfrac{\left(\pm 2 - M_n\right)\hat{n}_x a}{\gamma} \\[2mm] v + \dfrac{\left(\pm 2 - M_n\right)\hat{n}_y a}{\gamma} \\[2mm] \dfrac{a^2}{(\gamma+1)}\left(\dfrac{2}{(\gamma-1)} \pm 2M_n - M_n^2\right) + \dfrac{U^2}{2} \end{array}\right\} \tag{4.45}$$

let $\qquad \xi_0 = \pm\dfrac{\rho a(1 \pm M_n)^2}{4} \tag{4.46}$

and $\qquad \hat{E}^\pm(Q_n) = \left\{\begin{array}{c} u + \dfrac{\left(\pm 2 - M_n\right)\hat{n}_x a}{\gamma} \\[2mm] v + \dfrac{\left(\pm 2 - M_n\right)\hat{n}_y a}{\gamma} \\[2mm] \dfrac{a^2}{(\gamma+1)}\left(\dfrac{2}{(\gamma-1)} \pm 2M_n - M_n^2\right) + \dfrac{U^2}{2} \end{array}\right\} \tag{4.47}$

therefore the flux can be expressed as the product of the above terms:

$$E^{\pm}(\boldsymbol{Q}_n) = \xi_0 \hat{\boldsymbol{E}}^{\pm}(\boldsymbol{Q}_n) \tag{4.48}$$

By the application of the chain run:

$$\frac{\partial \boldsymbol{E}^{\pm}(\boldsymbol{Q})}{\partial \boldsymbol{Q}} = \frac{\partial \boldsymbol{E}^{\pm}(\boldsymbol{P})}{\partial \boldsymbol{P}} \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{Q}} \tag{4.49}$$

$$\frac{\partial \boldsymbol{E}^{\pm}(\boldsymbol{Q})}{\partial \boldsymbol{Q}} = \left[\xi_0 \frac{\partial \hat{\boldsymbol{E}}^{\pm}(\boldsymbol{P})}{\partial \boldsymbol{P}} + \hat{\boldsymbol{E}}^{\pm}(\boldsymbol{P}) \frac{\partial \boldsymbol{E}^{\pm}(\boldsymbol{P})}{\partial \boldsymbol{P}} \right] \frac{\partial \boldsymbol{P}}{\partial \boldsymbol{Q}} \tag{4.50}$$

where:

$$\boldsymbol{P} = \begin{Bmatrix} \rho \\ M_x \\ M_y \\ a \end{Bmatrix} \tag{4.51}$$

and

$$\frac{\partial \hat{\boldsymbol{E}}^{\pm}(\boldsymbol{P})}{\partial \boldsymbol{P}} = \begin{Bmatrix} 0 & 0 & 0 & 0 \\ 0 & a - \dfrac{a\hat{n}_x{}^2}{\gamma} & -\dfrac{a\hat{n}_x\hat{n}_y}{\gamma} & \dfrac{\zeta_1\hat{n}_x}{\gamma} + M_x \\ 0 & -\dfrac{a\hat{n}_x\hat{n}_y}{\gamma} & a - \dfrac{a\hat{n}_y{}^2}{\gamma} & \dfrac{\zeta_1\hat{n}_y}{\gamma} + M_y \\ 0 & \zeta_2\hat{n}_x + a^2 M_x & \zeta_2\hat{n}_y + a^2 M_y & \zeta_3 + aM_n^2 \end{Bmatrix} \tag{4.52}$$

where

$$\begin{aligned} \zeta_1 &= \pm 2 - M_n \\ \zeta_2 &= \frac{2a^2}{(\gamma+1)}\left(\pm 1 - M_n\right) \\ \zeta_3 &= \frac{2a}{(\gamma+1)}\left(\frac{2}{(\gamma-1)} \pm 2M_n - M_n^2\right) \end{aligned} \tag{4.53}$$

$$\zeta_4 = 1 \pm M_n \tag{4.54}$$

$$\frac{\partial \boldsymbol{E}^{\pm}(\boldsymbol{P})}{\partial \boldsymbol{P}} = \left\{ \pm \frac{\zeta_4{}^2}{2\rho} \quad \zeta_4\hat{n}_x \quad \zeta_4\hat{n}_y \quad \pm \frac{\zeta_4{}^2}{2a} \right\} \tag{4.55}$$

$$\frac{\partial \boldsymbol{P}}{\partial \boldsymbol{Q}} = \begin{Bmatrix} 1 & 0 & 0 & 0 \\ \dfrac{-u}{\rho a}\left(1+\dfrac{\rho \psi_0}{a}\right) & \dfrac{1}{\rho a}-\dfrac{u\psi_1}{a^2} & -\dfrac{u\psi_2}{a^2} & -\dfrac{u\psi_3}{a^2} \\ \dfrac{-v}{\rho a}\left(1+\dfrac{\rho \psi_0}{a}\right) & -\dfrac{v\psi_1}{a^2} & \dfrac{1}{\rho a}-\dfrac{v\psi_2}{a^2} & -\dfrac{v\psi_3}{a^2} \\ \psi_0 & \psi_1 & \psi_2 & \psi_3 \end{Bmatrix} \qquad (4.56)$$

$$\psi_0 = \frac{\gamma(\gamma-1)}{2\rho a}\left((u^2+v^2)-e_0\right)$$

$$\psi_1 = -\frac{u\gamma(\gamma-1)}{2\rho a}$$

$$\psi_2 = -\frac{v\gamma(\gamma-1)}{2\rho a} \qquad\qquad (4.57)$$

$$\psi_3 = \frac{\gamma(\gamma-1)}{2\rho a}$$

Finally the Jacobian $\dfrac{\partial \boldsymbol{E}^{\pm}(\boldsymbol{Q})}{\partial \boldsymbol{Q}}$ can be numerically constructed from the above matrices using equation (4.50).

4.7 Conservative form of the Navier-Stokes Equations

The conservative form of the non-dimensional Quasi-3D Navier-Stokes equations is:

$$\frac{\partial(h\boldsymbol{Q})}{\partial t} + div(h\boldsymbol{E}(\boldsymbol{Q})) = \frac{1}{Re}div(h\boldsymbol{F}(\boldsymbol{Q})) + \boldsymbol{H}_i(\boldsymbol{Q}) + \frac{1}{Re}\boldsymbol{H}_v(\boldsymbol{Q}) \qquad (4.37)$$

where h is the streamtube thickness, $\boldsymbol{F}(\boldsymbol{Q})$ is the viscous flux vector, and $\boldsymbol{H}_i(\boldsymbol{Q})$ and $\boldsymbol{H}_v(\boldsymbol{Q})$ are the inviscid and viscous components of the source term respectively.

It can be concluded from chapter 2 that an Euler study of the flow around a turbine blade is sufficient for the generation of an accurate pressure distribution, Xu & Denton (1990). For this reason, only the Euler solver has been fully validated. The full Navier-Stokes form has not been used in the present study and has subsequently not been validated. However, for completeness the full definition can be found in the appendix C.

4.8   **Concluding Remarks**

The governing equations and the computational method have been presented in this chapter, with various methods for solving the interface flux vector. The two most important methods: Roe's Approximate Riemann Solver and Van Veer's flux splitting, have been presented.

These have been used successfully in a number of publications to date. Examples are those by Currie & Carscallen (1998), Gehrer *et al* (2000), Furukawa *et al* (1992), and Zheng & Liu (1995) for flow through turbine cascades, Luo *et al* (1994) for detailed flow around the F18 Hornet, Frink (1994 & 1996) for his work on the ONERA M6 wing, and the computation of flow around NACA0012 aerofoils by Anderson (1992), and Zhang *et al* (1996).

5   NUMERICAL BOUNDARY CONDITIONS FOR THE EULER EQUATIONS

The limits of the computational domain must be correctly treated in order to accommodate the hyperbolic nature of the Euler equations. This is known as an IBVP (Initial Boundary Value Problem). Incorrect treatment would seriously affect the stability and accuracy of the computational solver. A number of physical constraints must be applied to the boundaries to ensure convergence. The number and type of these constraints depends on the physical conditions at the boundary, and the type of boundary in question. The Characteristic Boundary Method analyses the characteristic waves entering and leaving the domain and constrains a number of flow variables accordingly. The number of flow variables to be constrained is equal to the number of characteristic waves entering the domain. An easier technique is simply to extrapolate information from the interior of the domain to the boundary through zero, first or second order extrapolation.

There are several types of boundary associated with the current solver:

- Supersonic Inlet
- Subsonic Inlet
- Supersonic Outlet
- Subsonic Outlet
- Wall Boundary
- Periodic Boundary
- Fixed Boundary

5.1   Boundary Control Volumes

For the current study the following types of control volumes are presented. Two types of control volume are defined for the processing of information within the domain, *Flow* cells and *Boundary* cells. *Flow* cells are defined within the interior region of the domain and

always have nodes existing on the opposite side of the boundary faces, each of these faces can be considered to be a *Flow* face:
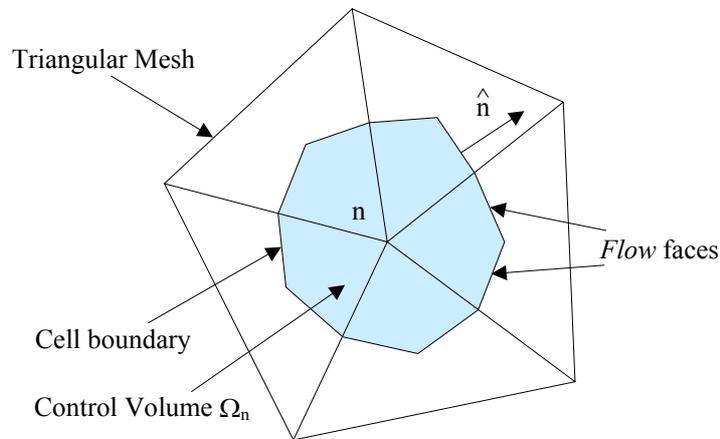
Figure 5.1: *Interior Control Volume*

Flow faces are processed using the Riemann solver previously described.

A *Boundary* cell on the other hand, will have boundary faces that do not have a corresponding node on the outside of the face, these are considered to be *Boundary* faces. Processing of these faces requires the direct evaluation of the flux $E(Q_B)$ at the boundary. This is based on the interior state and the fictitious exterior state dependent on the boundary type.
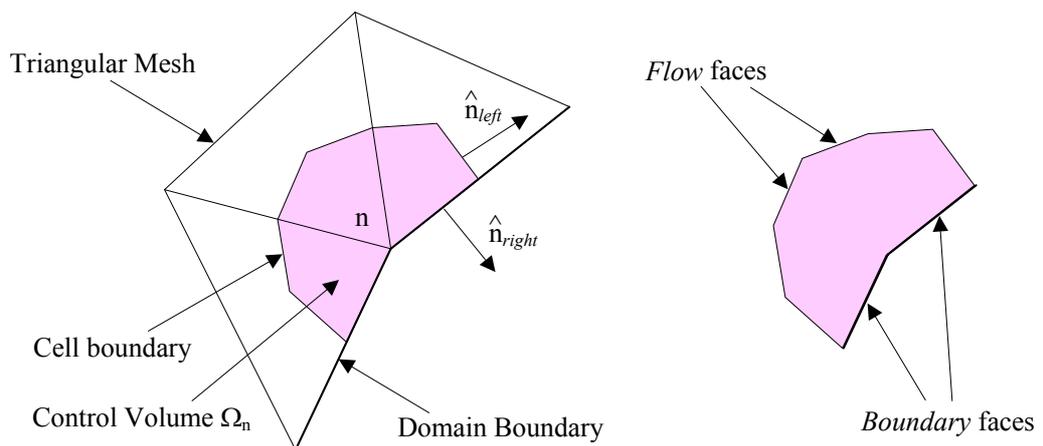
Figure 5.2: *Boundary Control Volume for Inlet, Outlet, and Fixed Boundaries*

There are three types of *Boundary* face associated with this type of control volume: *Inlet*, *Outlet* and *Fixed*.

In all boundary cases, the edge is split into two halves, $\hat{n}_{left}$ and $\hat{n}_{right}$. The left hand half of the face is treated in the normal fashion as a standard *Flow* edge as it is interior to the domain. The other half is treated according to the boundary type.

5.2 Supersonic Inlet Boundary

For a supersonic inlet, there is no influence on the boundary from the interior as all eigenvalues are positive into the domain. The inlet is expressed explicitly by a set of primitive variables, namely total temperature, Mach number, inlet flow angle and total pressure.

$$\boldsymbol{P}_B = \begin{Bmatrix} T_0 \\ M \\ \alpha \\ p_0 \end{Bmatrix} \qquad (5.1)$$

Using the isentropic relations:

$$\frac{T_0}{T} = 1 + \frac{(\gamma - 1)M^2}{2}$$

$$\frac{p_0}{p} = \left(\frac{T_0}{T}\right)^{\frac{\gamma}{\gamma - 1}} \qquad (5.2)$$

the conservative vector can then be constructed as follows:

$$T = \frac{T_0}{\left[1 + \frac{(\gamma - 1)M^2}{2}\right]}$$

$$a = \sqrt{\gamma R T}$$
$$u = \cos(\alpha)Ma$$
$$v = \sin(\alpha)Ma$$

$$p = p_0 \left(\frac{T}{T_0} \right)^{\frac{\gamma}{\gamma-1}}$$

$$\rho = \frac{p}{RT} \tag{5.3}$$

$$\rho e_0 = \frac{p}{(\gamma-1)} + \frac{\rho(u^2 + v^2)}{2}$$

to give:

$$\mathbf{Q}_B = \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ \rho e_0 \end{Bmatrix} \tag{5.4}$$

since the inlet is fully defined and constant, the derivative $\dfrac{\partial E(\mathbf{Q}_B)}{\partial \mathbf{Q}}$ will be zero.

5.3   Subsonic Inlet Boundary

For a subsonic inlet boundary condition, one of the eigenvalues at the inlet will be negative implying that one interior variable needs to be extrapolated and the others described by the inlet conditions. The inlet boundary primitive variables are again:

$$\mathbf{P}_B = \begin{Bmatrix} T_0 \\ M \\ \alpha \\ p_0 \end{Bmatrix} \tag{5.5}$$

but *M*, the Mach number is to be calculated from the required inlet Mach number and the interior Mach number, so as not to introduce any spurious waves into the solution. To ensure convergence and stability of the solution, the Mach number at the boundary is calculated by solving the equation, derived from the Van Leer Flux Splitting method:

$$\left[\frac{\rho a}{4} (1 + M_n)^2 \right]_I = \left[\frac{\rho a}{4} (1 + M_n)^2 \right]_B \tag{5.6}$$

The left hand side is completely described by the interior but the right hand side needs to be solved for the Mach number.

M is defined as the absolute Mach number in the direction of the inlet flow vector, M_n is the Mach number in the direction of the surface normal and M_x and M_y are the Mach numbers in the x and y directions.
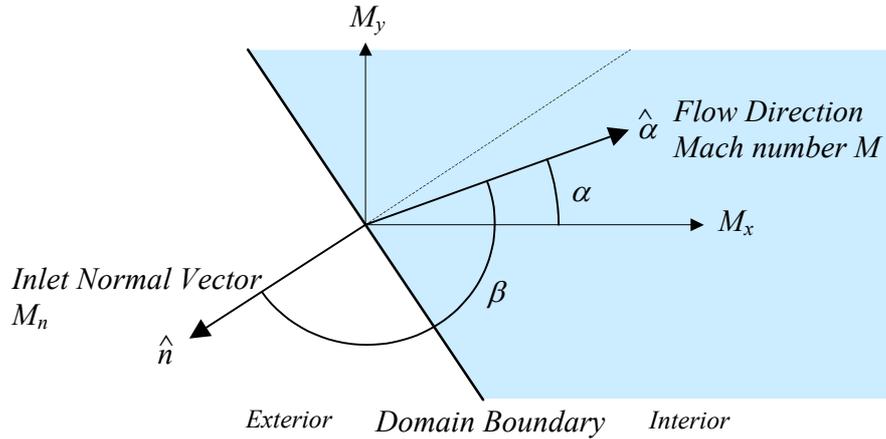
Figure 5.3: *Directions for Mach Numbers*

From the above diagram, M_x, M_y and M_n can be expressed as functions of M:

$$
\begin{aligned}
M_x &= M \cos(\alpha) \\
M_y &= M \sin(\alpha) \\
M_n &= M \cos(\beta)
\end{aligned}
\tag{5.7}
$$

If the flow direction is specified by the unit vector $\hat{\alpha} = \left(\alpha_x, \alpha_y\right)$ then $\cos(\beta)$ can be calculated from:

$$
\cos(\beta) = \hat{n} \cdot \hat{\alpha}
\tag{5.8}
$$

Through the isentropic relations, the interior Mach velocity and density can be expressed as functions of total temperature, total pressure and Mach number:

$$
\rho = \frac{p_0}{RT_0}\left(1 + \frac{(\gamma-1)M^2}{2}\right)^{-\frac{\gamma}{\gamma-1}}
\tag{5.9}
$$

$$a = \sqrt{\gamma R T_0}\left(1 + \frac{(\gamma - 1)M^2}{2}\right)^{-\frac{1}{2}} \tag{5.10}$$

therefore

$$\rho a = p_0 \sqrt{\frac{\gamma}{R T_0}}\left(1 + \frac{(\gamma - 1)M^2}{2}\right)^{-\frac{\gamma+1}{2(\gamma-1)}} \tag{5.11}$$

Substituting into (5.6) yields the equation:

$$\left(1 + M\cos(\beta)\right)^2\left(1 + \frac{(\gamma - 1)M^2}{2}\right)^{-\frac{\gamma+1}{2(\gamma-1)}} = \frac{\left[\rho a\left(1 + M_n\right)^2\right]_I}{\left[p_0\sqrt{\dfrac{\gamma}{R T_0}}\right]_B} \tag{5.12}$$

This can be solved for M using a Newton-Raphson algorithm (see appendix).

Once the Mach number M is known ρ and a can be recovered from (5.9) and (5.10), and p and T can be calculated from (5.2). The flux at the inlet is then defined as:

$$\boldsymbol{E}_B = \left\{ \begin{array}{c} \rho U \\ \rho u U + p\hat{n}_x \\ \rho v U + p\hat{n}_y \\ U\left(\dfrac{\gamma p}{(\gamma - 1)} + \dfrac{\rho(Ma)^2}{2}\right) \end{array} \right\} \tag{5.13}$$

where

$$U = \cos(\beta)Ma \qquad u = \cos(\alpha)Ma \qquad v = \cos(\alpha)Ma \qquad p = \rho R T \tag{5.14}$$

The implicit scheme also requires the definition of the Jacobian $\dfrac{\partial \boldsymbol{E}(\boldsymbol{Q})}{\partial \boldsymbol{Q}}$. This is evaluated

using the chain rule: $\dfrac{\partial \boldsymbol{E}(\boldsymbol{Q})}{\partial \boldsymbol{Q}} = \dfrac{\partial \boldsymbol{E}(\boldsymbol{P})}{\partial \boldsymbol{P}_B} \dfrac{\partial \boldsymbol{P}_B}{\partial \boldsymbol{Q}}$.

The inlet derivative is defined such that the Mach number is allowed to vary but all other

primitive variables are constrained. This is achieved by defining $\dfrac{\partial \boldsymbol{E}(\boldsymbol{P})}{\partial \boldsymbol{P}_B}$ and $\dfrac{\partial \boldsymbol{P}_B}{\partial \boldsymbol{Q}}$ by the

following method:

M is unconstrained, hence only those terms $\dfrac{\partial M}{\partial \boldsymbol{Q}}$ will be non-zero, i.e. the 2$^{\text{nd}}$ row of $\dfrac{\partial \boldsymbol{P}_B}{\partial \boldsymbol{Q}}$.

Expressing M in terms of $\boldsymbol{Q}$ gives:

$$M = \sqrt{\frac{Q_1^2 + Q_2^2}{\gamma(\gamma-1)\left(Q_0 Q_3 - \dfrac{\left(Q_1^2 + Q_2^2\right)}{2}\right)}} \tag{5.15}$$

therefore:

$$\frac{\partial \boldsymbol{P}_B}{\partial \boldsymbol{Q}} = \begin{Bmatrix} 0 & 0 & 0 & 0 \\ \phi_0 & \phi_1 & \phi_2 & \phi_3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{Bmatrix} \tag{5.16}$$

$$\phi_0 = -\frac{\zeta e_0 U}{2} \qquad \phi_1 = \frac{\zeta e_0 u}{U} \qquad \phi_2 = \frac{\zeta e_0 v}{U} \qquad \phi_3 = -\frac{\zeta U}{2} \tag{5.17}$$

where

$$\zeta = \frac{\gamma(\gamma-1)}{\rho a^3}$$

$$\rho = Q_0 \quad u = \frac{Q_1}{Q_0} \quad v = \frac{Q_2}{Q_0} \quad U = \sqrt{\frac{\left(Q_1^2 + Q_2^2\right)}{Q_0^2}} \quad e_0 = \frac{Q_3}{Q_0} \tag{5.18}$$

$$a = \sqrt{\frac{\gamma(\gamma-1)}{Q_0}\left(Q_3 - \frac{\left(Q_1^2 + Q_2^2\right)}{2Q_0}\right)}$$

Since only the 2nd row of $\dfrac{\partial \boldsymbol{P}_B}{\partial \boldsymbol{Q}}$ is defined, only the 2$^{\text{nd}}$ column of $\dfrac{\partial \boldsymbol{E}(\boldsymbol{P})}{\partial \boldsymbol{P}_B}$ needs defining:

$$\frac{\partial \boldsymbol{E}(\boldsymbol{P})}{\partial \boldsymbol{P}_B} = \begin{Bmatrix} 0 & \psi_0 & 0 & 0 \\ 0 & \psi_1 & 0 & 0 \\ 0 & \psi_2 & 0 & 0 \\ 0 & \psi_3 & 0 & 0 \end{Bmatrix} \tag{5.19}$$

where

$$\psi_0 = \frac{\rho U}{M}\left(\frac{1-M^2}{\xi_m}\right)$$

$$\psi_1 = -p_0\xi_m^{\left(\frac{1-2\gamma}{\gamma-1}\right)}\gamma M\Big(\big(\cos(\alpha)\cos(\beta)(M^2-2)\big)+\hat{n}_x\Big)$$

$$\psi_2 = -p_0\xi_m^{\left(\frac{1-2\gamma}{\gamma-1}\right)}\gamma M\Big(\big(\sin(\alpha)\cos(\beta)(M^2-2)\big)+\hat{n}_y\Big)$$

$$\psi_3 = \psi_0\frac{\gamma R T_0}{(\gamma-1)}$$

(5.20)

with

$$\xi_m = 1 + \frac{(\gamma-1)M^2}{2}$$

(5.21)

The derivative $\dfrac{\partial E(\boldsymbol{Q})}{\partial \boldsymbol{Q}}$ is then constructed from the elements:

$$\frac{\partial E_{[i,j]}(\boldsymbol{Q})}{\partial \boldsymbol{Q}} = \psi_i\phi_j \qquad\qquad \forall\ i,j = 1,\,5$$

(5.22)

5.4   Supersonic Outlet Boundary

For a supersonic outlet, all the eigenvalues are positive and passing out of the domain, therefore all boundary variables are taken from the interior. Hence $E(\boldsymbol{Q}_B) = E(\boldsymbol{Q}_I)$ and

$$\frac{\partial E(\boldsymbol{Q}_B)}{\partial \boldsymbol{Q}} = \frac{\partial E(\boldsymbol{Q}_I)}{\partial \boldsymbol{Q}}.$$

5.5   Subsonic Outlet Boundary

For a subsonic outlet condition, only one eigenvalue is negative implying that only one external variable has to be defined, all others being taken from the interior. In this case, the external downstream static pressure p_∞ is defined. The outlet boundary primitive variables are:

$$\boldsymbol{P}_B = \left\{\begin{array}{c} \rho \\ u \\ v \\ p_\infty \end{array}\right\}$$

(5.23)

The following method is introduced as a means of damping any spurious waves from entering the domain by modifying the boundary ρ, u and v variables.

The normal Mach number M_n at the boundary is given by the equation:

$$\alpha_3 M_n^3 + \alpha_2 M_n^2 + \alpha_1 M_n + \alpha_0 = 0 \tag{5.24}$$

where

$$
\begin{aligned}
\alpha_3 &= \gamma - 1 \\
\alpha_2 &= 2\gamma \\
\alpha_1 &= \gamma + 3 \\
\alpha_0 &= 2 - \frac{\left[\rho a (1 + M_n)^2 ((\gamma - 1)U + 2a)\right]_I}{\gamma p_\infty}
\end{aligned}
\tag{5.25}
$$

As with the subsonic inlet, this is solved using a Newton-Raphson algorithm to give M_n (see appendix). The Mach velocity at the boundary can then be recovered from:

$$a_B = \frac{\left[\gamma p_\infty (M_n + 1)^2\right]_B}{\left[pa(M_n + 1)^2\right]_I} \tag{5.26}$$

from which the normal velocity can be calculated:

$$[U_n]_B = \left[\frac{M_n}{a}\right]_B \tag{5.27}$$

The tangential velocity does not change, hence:

$$[U_t]_B = [U_t]_I \tag{5.28}$$

and the density can be extracted from:

$$\rho_B = \left[\frac{\gamma p_\infty}{a^2}\right]_B \tag{5.29}$$

Finally, the velocity has to be rotated back to the x, y co-ordinate system:

$$\begin{Bmatrix} u \\ v \end{Bmatrix} = \begin{Bmatrix} \hat{n}_x & -\hat{n}_y \\ \hat{n}_y & \hat{n}_x \end{Bmatrix} \begin{Bmatrix} U_n \\ U_t \end{Bmatrix} \tag{5.30}$$

For the implicit system, the Jacobian is defined by only applying a constraint to the pressure variable, the chain rule need not be applied and the Jacobian can be directly evaluated:

$$\frac{\partial E(Q)}{\partial Q} = \begin{Bmatrix} 0 & \hat{n}_x & \hat{n}_y & 0 \\ -uU_n & U_n + u\hat{n}_x & u\hat{n}_y & 0 \\ -vU_n & v\hat{n}_x & U_n + v\hat{n}_y & 0 \\ -U_n\left(h_0 + \dfrac{U^2}{2}\right) & h_0\hat{n}_x + uU_n & h_0\hat{n}_y + vU_n & 0 \end{Bmatrix} \tag{5.31}$$

5.6 Wall Boundary

There are two types of condition that can be enforced at a wall; these are the *slip-condition* and the *no-slip condition*, depending on whether an inviscid or viscous solution is desired respectively. In the inviscid case the fluid velocity is constrained to be tangential to the wall, ensuring that there is no mass flow across the wall. This is achieved by generating a node state in which the velocity is reflected about the direction of the wall, as shown in Figure 5.4.
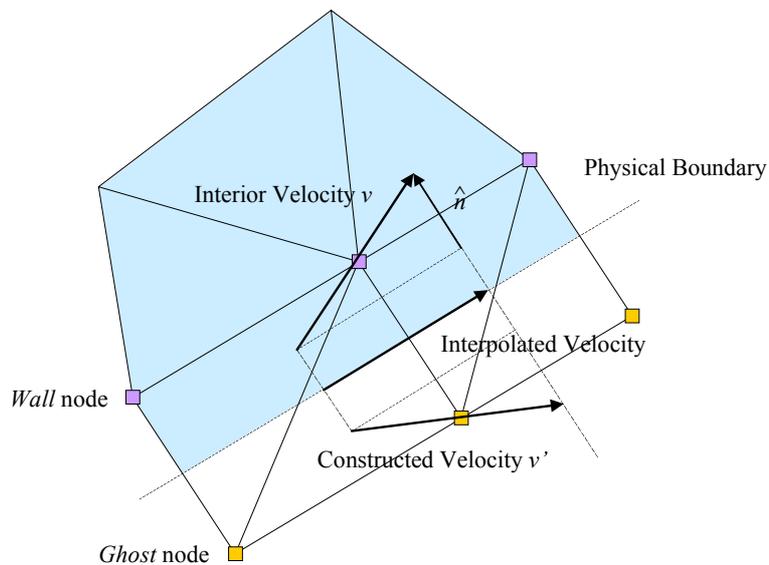

Figure 5.4: *Inviscid 'Slip' Wall Condition*

As the magnitude of the new velocity is the same, no other changes to density or total energy are required. Interpolation between the two gives a tangential velocity at the wall. In all schemes, exterior *Ghost* cells are used to define the boundary. The physical boundary is then the line half way between the defined *Wall* cells and the exterior *Ghost* cells.

The new velocity is constructed as follows:

$$v' = v - 2(\hat{n} \cdot v)\hat{n} \qquad\qquad\qquad (5.32)$$

where v is the interior velocity vector, v' is the constructed velocity vector and $\hat{n}$ is the unit normal vector of the wall.

5.7 Periodic Boundary

A *Periodic* boundary is defined in which opposite sides of the domain can be mapped onto each other. This effectively gives an infinite domain on one axis. Whenever an adjacent node is found to be periodic, the information is instead extracted from the corresponding cell on the opposite side of the domain. A turbine cascade consists of a series of identical blades arranged in a row, it would be computationally expensive to model all the blades considering that all the blades would give identical result. The periodic boundary minimises the memory and processing requirements by modelling the cascade as a single blade. The geometry for the cascade in the current study can be seen in chapter eleven.
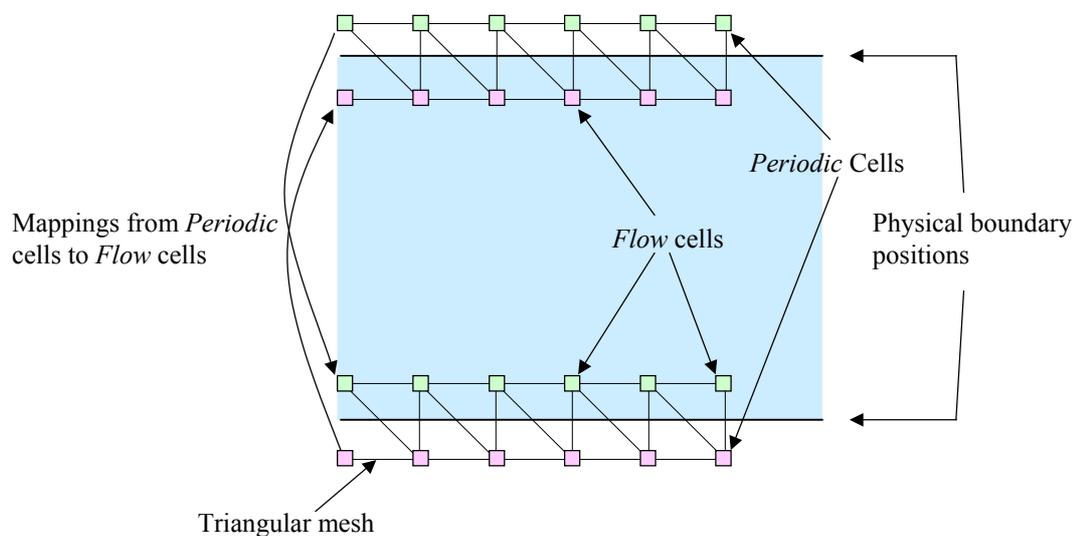

Figure 5.5: *Periodic Cell Mappings*

5.8   Fixed Boundary

The last type of boundary, a *Fixed* boundary, allows the exact defining of conditions along a particular edge. As with a supersonic inlet, all flow parameters are set explicitly allowing no flexibility. Great care must be taken as this may cause discontinuities in the flow and should only be used for the purposes of testing the code.

5.9   Numerical Boundary Conditions for the Navier-Stokes Equations

The presence of viscosity and heat transfer modifies the nature of the Euler equations. The resulting Navier-Stokes equations are now a mixed system of partial differential equations, (4.1), (4.2), and (4.3). The conservative laws for momentum and total energy are now second order. They are parabolic in time and elliptic in space and are known as *parabolic-elliptic*. The mass conservation equation, however, still remains hyperbolic.

A consequence of this is that a greater number of boundary conditions are now required. For subsonic inlets and outlets in three dimensions, according to Gustafsson and Sundström (1978), five and four conditions need to be imposed respectively. For the two-dimensional case this would reduce to four and three conditions. As it is the momentum and total energy terms that are no longer hyperbolic, it is these terms that require velocity and temperature information to be defined at the boundaries.

A reasonable approximation in such cases would be to assume that the flow at inlet and outlet boundaries is inviscid. Therefore the above processes for the Euler equations can be applied. It should be noted that this treatment would not correctly model flow behaviour, as it cannot generally be assumed that flow at the boundaries will be strictly inviscid. In particular, for the modelling of vortex shedding behind a cylinder, the wake flow will cross the exit plane, and viscous terms will still be present. However, the errors due to this approximation can be minimised if the outlet boundary is a sufficient distance from the cylinder, typically of the order of thirty or more cylinder diameters downstream. At this distance from the cylinder, the instabilities in the flow would have damped out considerably due to viscosity, resulting in a smooth flow. The loss of information due to the above approximation could be considered as negligible.

5.9.1   Solid Wall Boundary

The only boundary that needs to be considered for the Navier-Stokes equations is the flow along a solid wall. A similar method to the *slip* wall can be used with the modification that the extrapolated velocity in the ghost cell is now a reverse of the interior node, rather than a reflection about the wall normal vector. If the wall is assumed to be adiabatic then no other treatment is required. A zero pressure gradient is automatically imposed by a zero-order extrapolation of density and total energy from the interior, thus ensuring no mass-flow through the physical boundary. The new velocity *v'* is simply calculated as:

$$v' = -v \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \textbf{(5.33)}$$
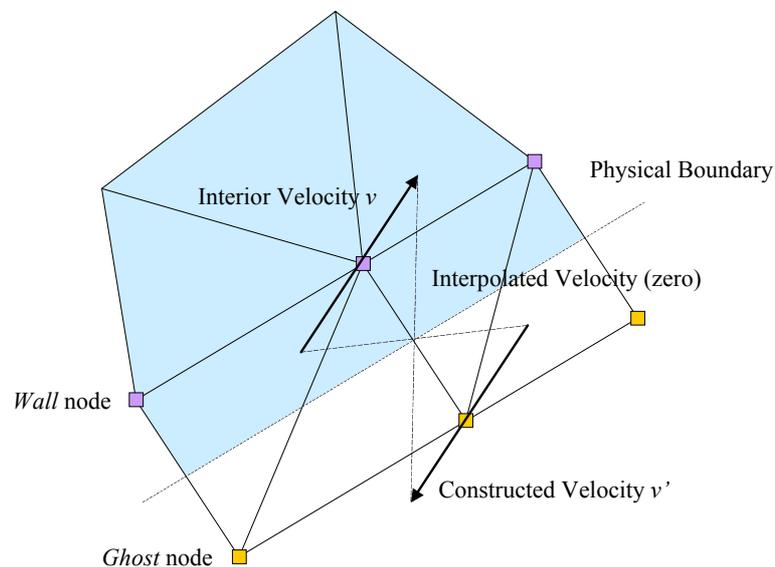
Figure 5.6: *Viscous 'No-Slip' Wall Condition*

5.10   Concluding Remarks

The relevant boundary conditions have been presented in this chapter. Of particular importance are the subsonic inlet and outlet conditions that have the advantage of minimising extraneous waves from being introduced into the domain. The inlet condition has the added advantage that the solution of the Mach number at the inlet, will not introduce numerical errors if the flow vector is close to the tangential vector of the outer boundary. This is particularly important if the outer boundary is circular or elliptical.

6 HIGHER ORDER UPWIND SCHEMES

All the schemes so far described have been first order accurate discretisations in space and so generate poor representations of the flow phenomena, even in simple cases, with the solutions containing far too much diffusion. It is therefore necessary to improve the resolution of the solver with higher order methods. Two types of upwind schemes have been examined for the current study, the MUSCL scheme (Monotone Upwind Schemes for Conservative Laws), named after the code first developed by Van Leer (1979), and non-MUSCL schemes in which the higher order is obtained through an operator embedded in the Riemann solver. The introduction of higher order schemes increases the error due to truncation of higher order terms making such schemes susceptible to oscillations near discontinuities in the flow; this is due to the larger stencil that is required to calculate the higher order terms. To ensure that the schemes do not produce unnecessary oscillations, i.e. they are *monotone*, no new extrema are introduced into the solution at any time step, the concept of a TVD (Total Variance Diminishing) scheme is introduced; originally conceived by Harten (1984). TVD schemes are obtained by applying some limiting function to the gradients of the dependent variables (slope limiters) or to the fluxes themselves (flux limiters). The Total Variance of a one-dimensional mesh is defined as:

$$TV(\boldsymbol{Q}^t) = \sum_i \left| \boldsymbol{Q}_{i+1} - \boldsymbol{Q}_i \right| \tag{6.1}$$

The scheme is said to be Total Variance Diminishing if at any time *t*:

$$TV(\boldsymbol{Q}^{t+1}) \leq TV(\boldsymbol{Q}^t) \tag{6.2}$$

i.e. at any time step the total variance is less than that of the previous time step - any new extrema would cause an increase. The scheme is *monotonicity preserving* if at any time no new extrema are introduced, and the values of local minima are non-decreasing and local maxima are non-increasing. The added property is maintained that if the scheme is monotone at time *t* then it will also be monotone at time *t*+1. Monotonicity is a necessary and sufficient condition for a TVD scheme; therefore all monotone schemes are TVD, and TVD schemes

preserve monotonicity. The concept of TVD would be achieved in the extrapolation cases if the gradients are constrained to ensure that the extrapolated values lies within some limits. The application of limiters will be described in each section.
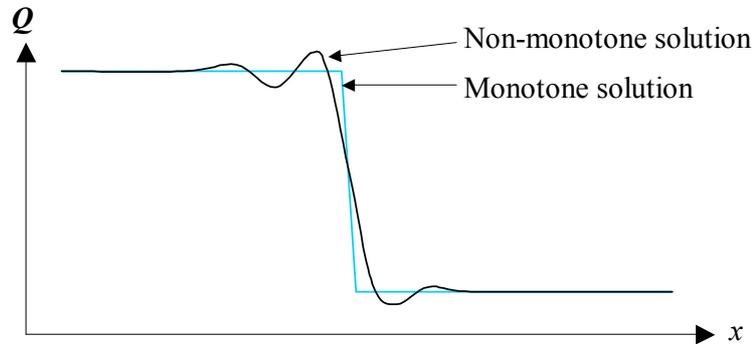
Figure 6.1: *Monotone and Non-Monotone Solutions*

6.1　　　MUSCL Scheme

The schemes so far described with the values taken at the centres of the control volumes are 1st order accurate in space. A higher order scheme is generated whereby the value of Q is extrapolated to the cell boundary depending on the gradient of Q within the cell. A one-dimensional example of this can be seen in Figure 6.2:
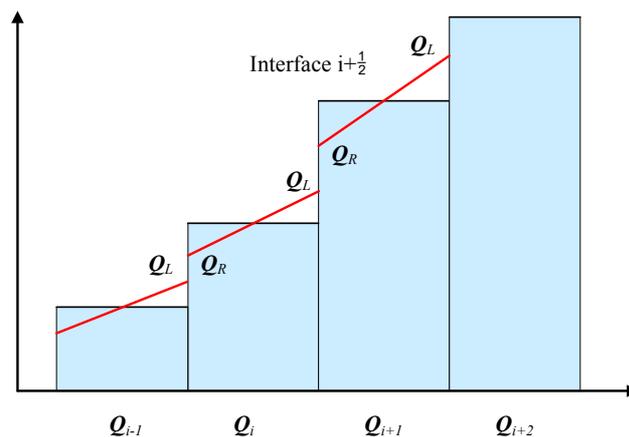
Figure 6.2: *MUSCL Extrapolation*

Using the one-dimensional example above, the values at the interface $i+\frac{1}{2}$ would be defined as:

$$Q_L = Q_i + \frac{1}{4}\left[(1-\kappa)\Delta Q_{i-\frac{1}{2}} + (1+\kappa)\Delta Q_{i+\frac{1}{2}}\right]$$

$$Q_R = Q_{i+1} - \frac{1}{4}\left[(1+\kappa)\Delta Q_{i+\frac{1}{2}} + (1-\kappa)\Delta Q_{i+\frac{3}{2}}\right]$$

$$\Delta Q_{i-\frac{1}{2}} = (Q_i - Q_{i-1}) \tag{6.3}$$

$$\Delta Q_{i+\frac{1}{2}} = (Q_{i+1} - Q_i)$$

$$\Delta Q_{i+\frac{3}{2}} = (Q_{i+2} - Q_{i+1})$$

where the parameter κ is used to define the order of the scheme as defined in the table below:

parameter κ	Description of scheme
-1	Second-order fully one-sided scheme
0	Linear interpolation based on upstream and downstream cell
$\frac{1}{2}$	Leonard's scheme
$-\frac{1}{3}$	Third-order upwind scheme
1	Central differencing scheme

Table 6.1: *Values for Parameter* κ

The one-dimensional scheme can be applied to two-dimensional unstructured meshes by extrapolation of the end points to give a four-point stencil. The scheme can then be applied in the method defined above. As seen in Figure 6.2, Q_n and Q_k are the original points; Q_{nn} and Q_{kk} are interpolated from within a triangle by the values at the vertices of that triangle:
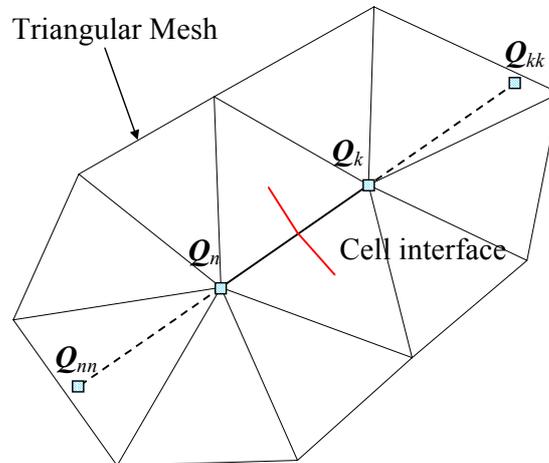
Figure 6.3: *Four-point Stencil*

6.1.1   Point Interpolation

The flow variables within a triangle are interpolated from the states at the vertices as defined in Figure 6.4, Foley *et al* (1990):
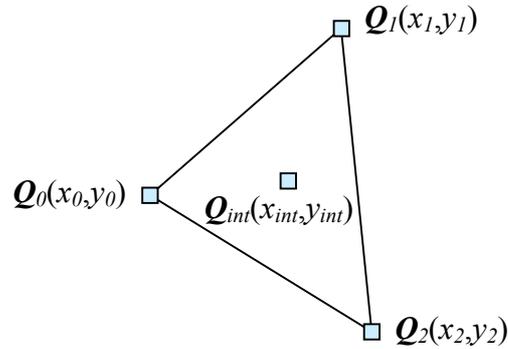
Figure 6.4: *Interpolation inside a Triangle*

Three weighting functions are defined for the three vertices, Q_0, Q_1, and Q_2:

$$denom = \begin{vmatrix} 1 & 1 & 1 \\ x_0 & x_1 & x_2 \\ y_0 & y_1 & y_2 \end{vmatrix} \qquad\qquad (6.4)$$

$$W_0 = \frac{\begin{vmatrix} 1 & 1 & 1 \\ x & x_1 & x_2 \\ y & y_1 & y_2 \end{vmatrix}}{denom} \qquad W_1 = \frac{\begin{vmatrix} 1 & 1 & 1 \\ x_0 & x & x_2 \\ y_0 & y & y_2 \end{vmatrix}}{denom} \qquad W_2 = \frac{\begin{vmatrix} 1 & 1 & 1 \\ x_0 & x_1 & x \\ y_0 & y_1 & y \end{vmatrix}}{denom} \qquad (6.5)$$

The interpolated interior state Q_{int} is then defined as:

$$Q_{int} = \sum_{i=0}^{2} W_i Q_i \qquad\qquad (6.6)$$

For those cases where the extrapolated co-ordinate lies outside the domain, a zero order extrapolation is used from the nearest Q_n or Q_k.

6.2   Minmod Limiter

The MUSCL scheme as described above is not enough to remove the appearance of oscillations in the solution when the scheme is extended to higher orders; Godunov (1959) shows that monotone schemes can be at most first order accurate. This necessitates the introduction of a *slope limiter* in the case of variable extrapolation; a non-linear corrective factor to recovery monotonicity in higher order schemes. The limiter used in the current study is based on the *Minmod* limiter, defined as:

$$minmod(a,b) = sign(a) \cdot MAX \left[0, \quad MIN \left(|a|, b \, sign(a) \right) \right] \tag{6.7}$$

The extrapolated values are limited by the introduction of a limiting function Φ. Applying to the $\boldsymbol{Q}_L$ term in (6.3) gives:

$$\boldsymbol{Q}_L = \boldsymbol{Q}_i + \frac{1}{4} \left[(1-\kappa) \Phi^+_{i-\frac{1}{2}} \Delta \boldsymbol{Q}_{i-\frac{1}{2}} + (1+\kappa) \Phi^-_{i+\frac{1}{2}} \Delta \boldsymbol{Q}_{i+\frac{1}{2}} \right] \tag{6.8}$$

where

$$\Phi^+_{i-\frac{1}{2}} = \Phi \left(r^+_{i-\frac{1}{2}} \right) \qquad \text{and} \qquad \Phi^-_{i+\frac{1}{2}} = \Phi \left(r^-_{i+\frac{1}{2}} \right) \tag{6.9}$$

These are forward and backward limiters applied to the upstream and downstream ratios:

$$r^+_{i-\frac{1}{2}} = \frac{\Delta \boldsymbol{Q}_{i+\frac{1}{2}}}{\Delta \boldsymbol{Q}_{i-\frac{1}{2}}} \qquad \text{and} \qquad r^-_{i+\frac{1}{2}} = \frac{\Delta \boldsymbol{Q}_{i-\frac{1}{2}}}{\Delta \boldsymbol{Q}_{i+\frac{1}{2}}} \tag{6.10}$$

A symmetry property is defined such that for a given limiter function:

$$\Phi(r) = r \Phi \left(\frac{1}{r} \right) \tag{6.11}$$

By replacing $r^+_{i-\frac{1}{2}}$ with r_L and rearranging (6.8), the following is derived:

$$\boldsymbol{Q}_L = \boldsymbol{Q}_i + \frac{1}{4} \left[(1-\kappa) \Phi(r_L) + (1+\kappa) r_L \Phi \left(\frac{1}{r_L} \right) \right] \Delta \boldsymbol{Q}_{i-\frac{1}{2}} \tag{6.12}$$

where

$$r_L = \frac{\Delta \boldsymbol{Q}_{i+\frac{1}{2}}}{\Delta \boldsymbol{Q}_{i-\frac{1}{2}}}$$

(6.13)

Equation (6.12) can now be expressed as:

$$\boldsymbol{Q}_L = \boldsymbol{Q}_i + \frac{1}{2}\Psi_L(r_L)\Delta \boldsymbol{Q}_{i-\frac{1}{2}}$$

(6.14)

where

$$\Psi_L(r_L) = \frac{1}{2}\left[(1-\kappa)\Phi(r_L) + (1+\kappa)r_L\Phi\left(\frac{1}{r_L}\right) \right]$$

(6.15)

Ψ_L is now the slope limiter. In order that the solution does in fact satisfy the TVD condition, the limiter must lie within some constraint. This is achieved by the constraints proposed by Sweby (1984):

$$1 \le \Psi \le \text{MIN}(2, r) \qquad\qquad \forall \, r > 1$$

$$r \le \Psi \le \text{MIN}(1, 2r) \qquad\qquad \forall \, 0 \le r \le 1$$

The limiter is therefore defined by the shaded area of Figure 6.5.
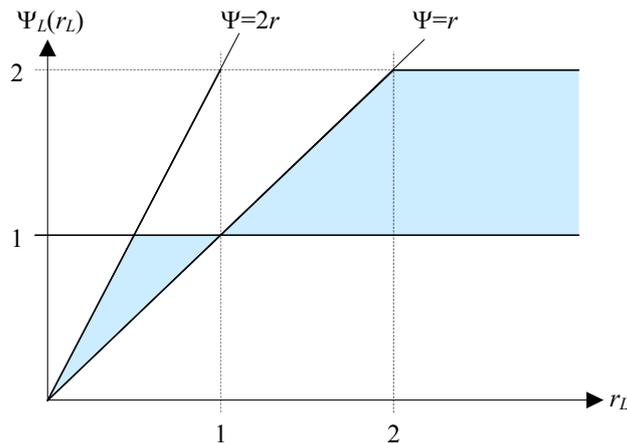
**Figure 6.5:** *Limiter Region for Second Order TVD Schemes*

Two such limiters are those of Van Leer (1974):

$$\Psi^{VL}(r) = \frac{r + |r|}{1 + r}$$

(6.16)

and the smoother limiter of Van Albada *et al* (1982):

$$\Psi^{VA}(r) = \frac{r^2 + r}{1 + r^2}$$

(6.17)

However, these have the disadvantage that they have the inconvenience of a ratio, and would require the addition of an ε term in the denominator to eliminate division by zero errors. An alternative to this is the *minmod* limiter of the type used in the current study. This limiter is applied to the ΔQ's from (6.3) in the following manner:

$$Q_L = Q_i + \frac{1}{4}\left[(1-\kappa)\Delta\overline{Q}_{i-\frac{1}{2}} + (1+\kappa)\Delta\overline{\overline{Q}}_{i+\frac{1}{2}}\right]$$

$$Q_R = Q_{i+1} - \frac{1}{4}\left[(1+\kappa)\Delta\overline{Q}_{i+\frac{1}{2}} + (1-\kappa)\Delta\overline{\overline{Q}}_{i+\frac{3}{2}}\right]$$

$$\Delta\overline{Q}_{i-\frac{1}{2}} = \text{minmod}\left(\Delta Q_{i-\frac{1}{2}}, \quad \omega\Delta Q_{i+\frac{1}{2}}\right)$$

$$\Delta\overline{\overline{Q}}_{i+\frac{1}{2}} = \text{minmod}\left(\Delta Q_{i+\frac{1}{2}}, \quad \omega\Delta Q_{i-\frac{1}{2}}\right)$$

(6.18)

$$\Delta\overline{Q}_{i+\frac{1}{2}} = \text{minmod}\left(\Delta Q_{i+\frac{1}{2}}, \quad \omega\Delta Q_{i+\frac{3}{2}}\right)$$

$$\Delta\overline{\overline{Q}}_{i+\frac{3}{2}} = \text{minmod}\left(\Delta Q_{i+\frac{3}{2}}, \quad \omega\Delta Q_{i+\frac{1}{2}}\right)$$

where

$$1 \le \omega \le \frac{3-\kappa}{1-\kappa}$$

(6.19)

6.3   Barth MUSCL Scheme

An alternative approach by Barth & Jesperson (1989) is to calculate the gradient by Green-Gauss reconstruction:

$$\nabla Q_n = \frac{1}{A_n}\sum_{f \in F_n}\left(Q_f \hat{n}\right)$$

(6.20)

where F_n is set of boundary faces for cell n, A_n is the area of the cell, $\hat{n}$ is the normal vector to the surface, and Q_f is the value of Q at the midpoint of the face.
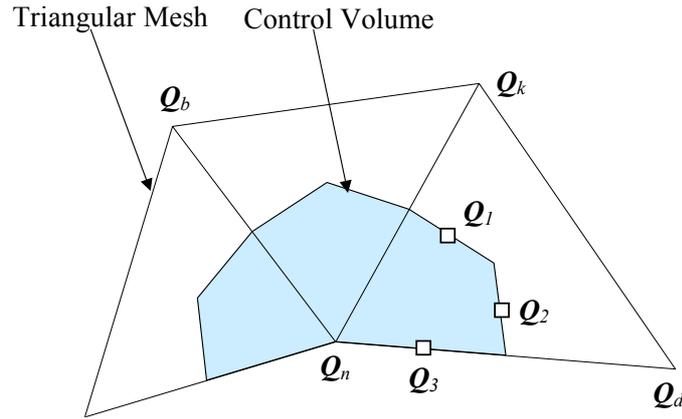
Figure 6.6: *Gradient Calculation*

The values of $\boldsymbol{Q}_f$ are calculated by 1st order interpolation from the neighbouring node values in two different ways depending on the face type, as in Figure 6.6:

- The centre of a standard *Flow* face: $\quad \boldsymbol{Q}_1 = \dfrac{5\boldsymbol{Q}_n + 5\boldsymbol{Q}_k + 2\boldsymbol{Q}_d}{12}$

- A *Flow* face associated with a boundary edge: $\quad \boldsymbol{Q}_2 = \dfrac{5\boldsymbol{Q}_n + 2\boldsymbol{Q}_k + 5\boldsymbol{Q}_d}{12}$

- A *Boundary* face associated with a boundary edge: $\boldsymbol{Q}_3 = \dfrac{3\boldsymbol{Q}_n + \boldsymbol{Q}_d}{4}$

The value of $\boldsymbol{Q}_n$ and $\boldsymbol{Q}_k$ can now be extrapolated to the cell interface using the cell gradient and the vector to the middle of face f :

$$\begin{aligned}
\boldsymbol{Q}_L &= \boldsymbol{Q}_n + \nabla \boldsymbol{Q}_n \cdot \hat{v}_f \\
\boldsymbol{Q}_R &= \boldsymbol{Q}_k + \nabla \boldsymbol{Q}_k \cdot \hat{w}_f
\end{aligned} \qquad\qquad \textbf{(6.21)}$$

where $\hat{w}_f$ is the vector from the adjacent node to the midpoint of face f as shown in Figure 6.7:

Control Volume
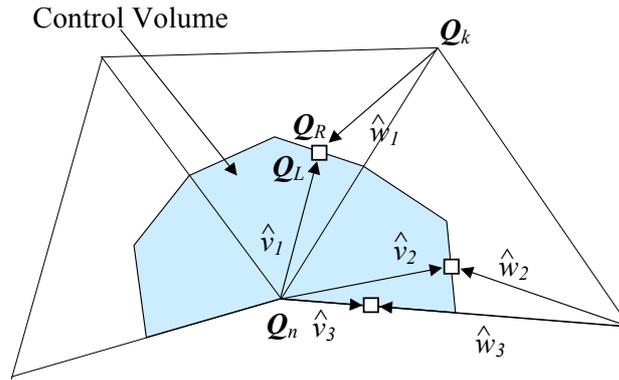
Figure 6.7: *Extrapolated $\mathbf{Q}_L$ and $\mathbf{Q}_R$*

The minmod limiter Φ_n is again applied, but in this scheme it is applied radially around the central node; hence the formulation is slightly different:

$$\mathbf{Q}_f = \mathbf{Q}_n + \nabla\mathbf{Q}_n \cdot \hat{v}_f$$

$$\Phi_n = \begin{cases} MIN\left(1, \dfrac{\mathbf{Q}_n^{Max} - \mathbf{Q}_f}{\mathbf{Q}_n - \mathbf{Q}_f}\right) & \mathbf{Q}_n - \mathbf{Q}_f > 0 \\[2mm] 1 & \mathbf{Q}_n - \mathbf{Q}_f = 0 \qquad \forall f \in F_n \\[2mm] MIN\left(1, \dfrac{\mathbf{Q}_n^{Min} - \mathbf{Q}_f}{\mathbf{Q}_n - \mathbf{Q}_f}\right) & \mathbf{Q}_n - \mathbf{Q}_f < 0 \end{cases} \tag{6.22}$$

where

$$\begin{aligned} \mathbf{Q}_n^{Max} &= MAX(\mathbf{Q}_n, \mathbf{Q}_k) \\ \mathbf{Q}_n^{Min} &= MIN(\mathbf{Q}_n, \mathbf{Q}_k) \end{aligned} \qquad \forall k \in \text{neighbours of } n \tag{6.23}$$

The effect of this limiter can be seen in the diagram below.
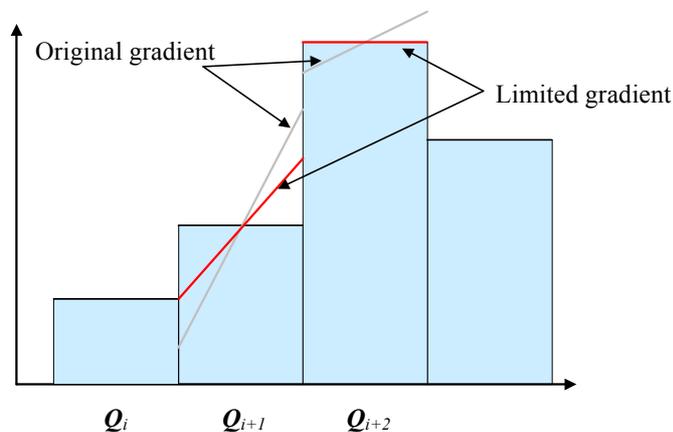
Figure 6.8: *Applying a Limiter*

The extrapolated values can now be calculated with the new limiter with the following equation:

$$\boldsymbol{Q}_L = \boldsymbol{Q}_n + \Phi_n \nabla \boldsymbol{Q}_n \cdot \hat{v}_f$$
$$\boldsymbol{Q}_R = \boldsymbol{Q}_k + \Phi_k \nabla \boldsymbol{Q}_k \cdot \hat{w}_f$$

(6.24)

which are now used in place $\boldsymbol{Q}_n$ and $\boldsymbol{Q}_k$ when evaluating the inviscid flux. Both the methods described above can be applied to either Roe's scheme or to Van Leer's flux splitting scheme.

6.4 Embedded Scheme: Flux Limiter

In an embedded scheme an operator is introduced into the calculation of the Riemann solver, Quirk (1990); in this manner, each of the characteristic waves is upwinded individually. The explicit part of the Roe-averaged Matrix is now calculated as:

$$\left|\widetilde{A}\right|(\boldsymbol{Q}_k - \boldsymbol{Q}_n) = \widetilde{T}\left|\widetilde{\Lambda}\right|[I - \Psi]\widetilde{T}^{-1}\Delta\boldsymbol{Q}$$

(6.25)

where Ψ is the matrix whose diagonal elements consist of the limiters based on parameters r_i, the ratios of the upstream to local wave strengths:

$$\Psi_i = LIMITER(r_i) \quad where \quad r_i = \frac{(\alpha_i)_{upwind}}{(\alpha_i)_{local}}$$

(6.26)

Looking back at the definition of Roe's Riemann Solver, the local wave strengths are given by the column vector:

$$\alpha_{local} = \widetilde{T}^{-1}\Delta\boldsymbol{Q}$$

(6.27)

The upstream wave strength vector is given by the Roe-averaged quantities, based on either $\boldsymbol{Q}_n$ and $\boldsymbol{Q}_{nn}$ if $\widetilde{\lambda}_i$ is positive or $\boldsymbol{Q}_k$ and $\boldsymbol{Q}_{kk}$ if $\widetilde{\lambda}_i$ is negative.

The division in the calculation of r_i can be removed by calculating each wave strength in turn and expanding out the term $[1 - \Psi_i](\alpha_i)_{local}$ to give:

$$[1 - \Psi_i](\alpha_i)_{local} = (\alpha_i)_{local} - LIMITER\big((\alpha_i)_{local}, \quad (\alpha_i)_{upwind}\big)$$

(6.28)

For the current study, the most efficient limiter was found by using the *minmod* limiter in the following form:

$$LIMITER(a, \quad b) = \frac{2}{3}\text{minmod}(a,3b) + \text{minmod}(a,\frac{b}{3}) \tag{6.29}$$

Examination of the following graph shows that this is similar to the limiter of Van Leer (6.17):
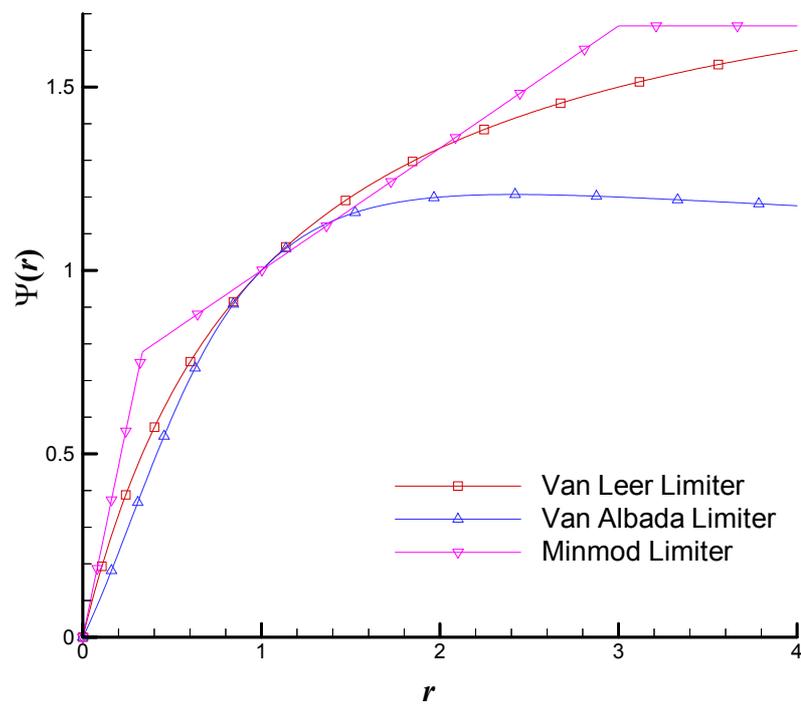
Figure 6.9: *Van Leer, Van Albada, and Minmod Limiters*

Higher values of ω in the limiter increase the resolution of shocks at a slight detriment to the monotonicity of the scheme. It should be noted that the 1st order scheme is fully monotone.

6.5   Concluding Remarks

The upwind schemes presented here constitute all those examined in the present study. Of the schemes investigated, the minmod limiter described in equation (6.29) was chosen as the best compromise of embedded limiters for the current study.

7 TIME INTEGRATION

Time Integration is performed using three different methods: Explicit steady state, 1st order Implicit and 2nd order Implicit. All methods are derived from (4.14) and (4.16):

$$\frac{\Delta \boldsymbol{Q}_n}{\Delta t} = -\frac{1}{2A_n} \sum_{k=1}^{d_n} \left\{ \left(\boldsymbol{E}(\boldsymbol{Q}_L) + \boldsymbol{E}(\boldsymbol{Q}_R) \right) \cdot \hat{n} - \left| \widetilde{A} \right| (\boldsymbol{Q}_R - \boldsymbol{Q}_L) \right\} \qquad (7.1)$$

In the *explicit formulation*, the terms on the right hand side of (7.1) are evaluated at time t and those on the left hand side at time $t+1$. Since all the information at time step t is always available at every iteration, the equation can be evaluated in a single operation. However, for the *implicit formulation*, both sides of the equation are evaluated at time $t+1$. The information on the right hand side is not fully defined, and the equation must be iterated to a solution.

7.1 Explicit Steady State

For the explicit method, the right hand side of the above equation is first denoted by $-\boldsymbol{R}(\boldsymbol{Q})$, the discrete form of the equation can then be written as:

$$\boldsymbol{R}(\boldsymbol{Q}) = -\frac{\Delta \boldsymbol{Q}_n}{\Delta t} \qquad (7.2)$$

Time integration of the equations is performed with an Explicit Four-Stage Runge-Kutta algorithm, first introduced by Jameson *et al* (1981), with local time-stepping to give $\boldsymbol{Q}_n$ at the next time step:

$$
\begin{aligned}
\boldsymbol{Q}_n^{(0)} &= \boldsymbol{Q}_n^t \\
\boldsymbol{Q}_n^{(1)} &= \boldsymbol{Q}_n^{(0)} - \alpha_1 \Delta t \boldsymbol{R}(\boldsymbol{Q}_n^{(0)}) \\
\boldsymbol{Q}_n^{(2)} &= \boldsymbol{Q}_n^{(0)} - \alpha_2 \Delta t \boldsymbol{R}(\boldsymbol{Q}_n^{(1)}) \\
\boldsymbol{Q}_n^{(3)} &= \boldsymbol{Q}_n^{(0)} - \alpha_3 \Delta t \boldsymbol{R}(\boldsymbol{Q}_n^{(2)}) \\
\boldsymbol{Q}_n^{(4)} &= \boldsymbol{Q}_n^{(0)} - \alpha_4 \Delta t \boldsymbol{R}(\boldsymbol{Q}_n^{(3)}) \\
\boldsymbol{Q}_n^{t+1} &= \boldsymbol{Q}_n^{(4)}
\end{aligned}
\qquad (7.3)
$$

where

$$\alpha_1 = \frac{1}{4} \quad \alpha_2 = \frac{1}{3} \quad \alpha_3 = \frac{1}{2} \quad \alpha_4 = 1 \tag{7.4}$$

The maximum allowable time step for any given cell is defined as:

$$\Delta t = \frac{\sqrt{A_n}}{\left(|U| + a\right)} \times CFL \tag{7.5}$$

where A_n is the area of the cell, a is the local speed of sound, $|U|$ is the local velocity, and CFL is the Courant-Friedrich-Levy number. A CFL number of 1 inserted into (7.5) will give the time taken for the fastest characteristic wave to propagate across the given cell.

This is a stationary method, meaning that information is only gained from the immediate neighbouring nodes, and as such has several disadvantages; namely that to ensure the stability of the solution, the maximum CFL number is defined as substantially less than one. Secondly, the rate of convergence to a solution is low due to the CFL constraint, with the accuracy of the residual only decreasing by 1 or 2 orders of magnitude.

7.2   1st Order Implicit Method

In theory, *Implicit* time integration methods are unconditionally stable, implying that they are stable, independent of the time step, effectively removing the CFL constraint. However this is only true for linear systems. In practice, the time step must still be restricted for non-linear systems with the CFL value begin dependent on the particular solver being used. In some cases, the CFL number may even be close to values for explicit solvers as described previously.

For implicit time integration a different approach is required. All the terms on the right hand side of equation (3.17) are first defined at time step *t*+1, Currie & Carscallen (1998), Arnone *et al* (1995), to give:

$$\frac{\Delta \boldsymbol{Q}_n}{\Delta t} = -\frac{1}{2A_n} \sum_{k=1}^{d_n} \left\{ \left(\boldsymbol{E}(\boldsymbol{Q}_L^{t+1}) + \boldsymbol{E}(\boldsymbol{Q}_R^{t+1}) \right) \cdot \hat{n} - \left| \widetilde{A} \right| \left(\boldsymbol{Q}_R^{t+1} - \boldsymbol{Q}_L^{t+1} \right) \right\} \tag{7.6}$$

where the value of $\boldsymbol{Q}$ at time $t+1$ is given by:

$$\boldsymbol{Q}_n^{t+1} = \boldsymbol{Q}_n^t + \Delta\boldsymbol{Q}_n \qquad (7.7)$$

In order for the equation to be solved, the values on the right hand side must be evaluated at time t. This is accomplished by using a Taylor's Series expansion of each of the terms about time t. For the first term $\boldsymbol{E}(\boldsymbol{Q}_L^{t+1})$ this gives:

$$\boldsymbol{E}(\boldsymbol{Q}_L^{t+1}) = \boldsymbol{E}(\boldsymbol{Q}_L^t) + \frac{\partial\boldsymbol{E}(\boldsymbol{Q}_L^t)}{\partial t}.\Delta t \qquad (7.8)$$

the chain rule $\dfrac{\partial y}{\partial t} = \dfrac{\partial y}{\partial x}\dfrac{\partial x}{\partial t}$ can then be applied:

$$
\begin{aligned}
\boldsymbol{E}(\boldsymbol{Q}_L^{t+1}) &= \boldsymbol{E}(\boldsymbol{Q}_L^t) + \frac{\partial\boldsymbol{E}(\boldsymbol{Q}_L^t)}{\partial\boldsymbol{Q}}.\frac{\partial\boldsymbol{Q}}{\partial t}\Delta t \\
&= \boldsymbol{E}(\boldsymbol{Q}_L^t) + \frac{\partial\boldsymbol{E}(\boldsymbol{Q}_L^t)}{\partial\boldsymbol{Q}}.\frac{\Delta\boldsymbol{Q}}{\Delta t}\Delta t \qquad (7.9) \\
&= \boldsymbol{E}(\boldsymbol{Q}_L^t) + \frac{\partial\boldsymbol{E}(\boldsymbol{Q}_L^t)}{\partial\boldsymbol{Q}_n}.\Delta\boldsymbol{Q}_n
\end{aligned}
$$

Similarly for all the other terms on the right hand side:

$$\boldsymbol{E}(\boldsymbol{Q}_R^{t+1}) = \boldsymbol{E}(\boldsymbol{Q}_R^t) + \frac{\partial\boldsymbol{E}(\boldsymbol{Q}_R^t)}{\partial\boldsymbol{Q}_k}.\Delta\boldsymbol{Q}_k \qquad (7.10)$$

$$\boldsymbol{Q}_L^{t+1} = \boldsymbol{Q}_L^t + \Delta\boldsymbol{Q}_n \qquad (7.11)$$

$$\boldsymbol{Q}_R^{t+1} = \boldsymbol{Q}_R^t + \Delta\boldsymbol{Q}_k \qquad (7.12)$$

The only exception to the above is the Roe matrix $\left|\tilde{A}\right|$, which is left at time step t to minimise overheads in processing time and memory storage. The derivations for the Jacobian matrices are quite large and complex and are given in Appendix C. Substituting for the above terms into equation (3.19) and rearranging so that the $\Delta\boldsymbol{Q}_n$, $\Delta\boldsymbol{Q}_k$, and $\Delta\boldsymbol{Q}_{k^+}$ terms are on the left hand side gives:

$$\left[\frac{C_n \boldsymbol{I}}{\Delta t} + \boldsymbol{A}_n^t\right]\Delta \boldsymbol{Q}_n + \sum_{k=1}^{d_n}\left\{\boldsymbol{B}_{n,k}^t \Delta \boldsymbol{Q}_k\right\} = \boldsymbol{R}_n^t \tag{7.13}$$

where

$$\boldsymbol{A}_n^t = -\sum_{k=1}^{d_n}\left\{\frac{\partial \boldsymbol{E}_x(\boldsymbol{Q}_L^t)}{\partial \boldsymbol{Q}_n}\hat{n}_x + \frac{\partial \boldsymbol{E}_y(\boldsymbol{Q}_L^t)}{\partial \boldsymbol{Q}_n}\hat{n}_y + \left|\tilde{A}\right|\right\}l \tag{7.14}$$

$$\boldsymbol{B}_{n,k}^t = -\left\{\frac{\partial \boldsymbol{E}_x(\boldsymbol{Q}_R^t)}{\partial \boldsymbol{Q}_k}\hat{n}_x + \frac{\partial \boldsymbol{E}_y(\boldsymbol{Q}_R^t)}{\partial \boldsymbol{Q}_k}\hat{n}_y - \left|\tilde{A}\right|\right\}l \tag{7.15}$$

$$\boldsymbol{R}_n^t = \sum_{k=1}^{d_n}\left\{\left(\boldsymbol{E}(\boldsymbol{Q}_L^t) + \boldsymbol{E}(\boldsymbol{Q}_R^t)\right)\cdot\hat{n} - \left|\tilde{A}^t\right|(\boldsymbol{Q}_R^t - \boldsymbol{Q}_L^t)\right\}l \tag{7.16}$$

and

$$C_n = -2A_n \tag{7.17}$$

The x and y subscripts denote the x and y components of the Jacobian matrices. n is the central node, and k is an adjacent node.

The above set of equations when applied to all the relevant nodes in the system now defines a complete set of simultaneous equations in $\Delta \boldsymbol{Q}_i$. This is called a *Newton Linearisation* and defines a set of linear equations. With the aid of an example node, the system can be expressed in matrix form as follows:

Given an arbitrary *Flow* node with a set of 5 neighbours:
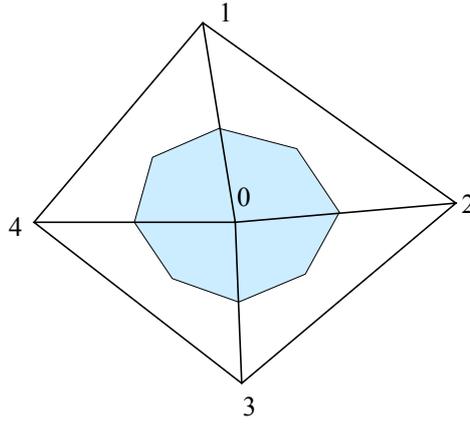
Figure 7.1: *Example Node for Matrix Structure*

Placing the above nodes into the matrix gives:

$$
\left\{
\begin{array}{ccccc}
\left[\dfrac{C_0 I}{\Delta t} - A_0^t\right] & B_{0,1}^t & B_{0,2}^t & B_{0,3}^t & B_{0,4}^t \\[3mm]
B_{1,0}^t & \left[\dfrac{C_1 I}{\Delta t} - A_1^t\right] & 0 & 0 & 0 \\[3mm]
B_{2,0}^t & 0 & \left[\dfrac{C_2 I}{\Delta t} - A_2^t\right] & 0 & 0 \\[3mm]
B_{3,0}^t & 0 & 0 & \left[\dfrac{C_3 I}{\Delta t} - A_3^t\right] & 0 \\[3mm]
B_{4,0}^t & 0 & 0 & 0 & \left[\dfrac{C_4 I}{\Delta t} - A_4^t\right]
\end{array}
\right\}
\left\{
\begin{array}{c}
\Delta Q_0 \\ \Delta Q_1 \\ \Delta Q_2 \\ \Delta Q_3 \\ \Delta Q_4
\end{array}
\right\}
=
\left\{
\begin{array}{c}
R_0 \\ R_1 \\ R_2 \\ R_3 \\ R_4
\end{array}
\right\}
$$

$$\tag{7.18}$$

This method is also termed a 'Steady state' solver due to the time integration being only 1st order. As such, it can be used for time dependent solutions, but with limitations applied to the time step.

7.3   2nd Order Implicit Method

The 1st order method has been previously defined as:

$$
\left[\frac{I}{\Delta t} - \frac{\partial R}{\partial Q}\right]_n^t \Delta Q_n^t = R_n^t
\tag{7.19}
$$

where

$$\boldsymbol{Q}_n^{t+1} = \boldsymbol{Q}_n^t + \Delta\boldsymbol{Q}_n^t \qquad\qquad\qquad (7.20)$$

A higher order time accurate scheme is then devised by defining the residual at the next time step, Currie & Carscallen (1998), Arnone *et al* (1995), to give:

$$\frac{3\boldsymbol{Q}_n^{t+1} - 4\boldsymbol{Q}_n^t + \boldsymbol{Q}_n^{t-1}}{2\Delta t} = \boldsymbol{R}_n^{t+1} \qquad\qquad (7.21)$$

the unsteady equation then becomes:

$$\left[\frac{3}{2}\frac{\boldsymbol{I}}{\Delta t} - \frac{\partial\boldsymbol{R}}{\partial\boldsymbol{Q}}\right]_n^{t+1,k} \Delta\boldsymbol{Q}_n^{t+1,k} = \boldsymbol{R}_n^{t+1,k} - \left\{\frac{3\boldsymbol{Q}_n^{t+1,k} - 4\boldsymbol{Q}_n^t + \boldsymbol{Q}_n^{t-1}}{2\Delta t}\right\} \qquad (7.22)$$

where

$$\boldsymbol{Q}_n^{t+1,k+1} = \boldsymbol{Q}_n^{t+1,k} + \Delta\boldsymbol{Q}_n^{t+1,k} \qquad\qquad (7.23)$$

The subscript *n* denotes the equation for a given node, *t* is the current time-step, *t+1* is the next time-step, and *k* is the *k*th sub-iteration to the next time-step.

A 1st order backward predictor is used to start the inner iterations with:

$$\boldsymbol{Q}_n^{t+1,0} = \boldsymbol{Q}_n^t + \left\{\frac{\boldsymbol{Q}_n^t - \boldsymbol{Q}_n^{t-1}}{2}\right\} \qquad\qquad (7.24)$$

7.4   Concluding Remarks

Various time stepping methods have been presented in this chapter that are all independent of the spatial discretisation. Explicit methods have the advantage of their reduced computational costs over the implicit methods. However, the effect of the stability limit on the allowable time-step can restrict their use to the simulation of low frequency events or very small time-steps. If the time step of the physical phenomena to be investigated is far larger than the stability limit allows, then implicit schemes may still be beneficial. It should also be noted that the convergence rate to a steady state solution of an implicit scheme is superior to that of an explicit scheme.

8   MATRIX INVERSION

As shown from the previous section, at every iteration a linear system of the type *Ax=b* must be solved. In this case, the *x* will be the vector of Δ***Q***s used for updating the conservative variables to the next time step. Due to the processing time and memory requirements such a system poses, iterative methods are normally employed. In this study, the system is solved by GMRES, a Krylov sub-space method with a Block-ILU preconditioner, Saad & Schultz (1986). To reduce memory requirements, the matrix is stored in sparse format, in which only the non-zero elements are defined. The preconditioner transforms the system into the form *MAx=Mb*, if *M=A⁻¹*, this implies a direct inversion of the matrix that will solve the system within a single operation. However, this is impractical due to the limitations previously stated, justifying the use of the preconditioner.

The efficiency of the algorithm relies heavily on the arrangement of the information within the matrix to be solved. A measure of the effort required can be estimated by the bandwidth of the matrix, which is defined as the distance to the furthest non-zero element from the diagonal line:
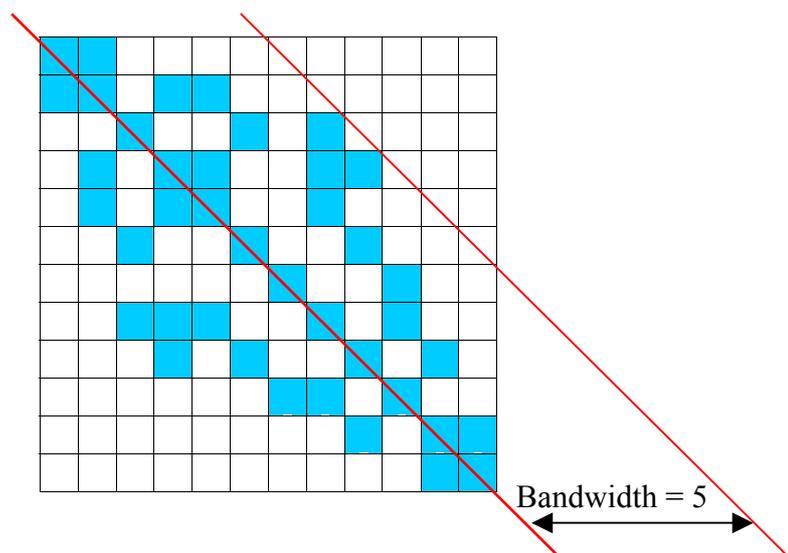

Figure 8.1*: Matrix Bandwidth*

In this study, a Cuthill-McKee algorithm, Cuthill & McKee (1969), is employed to generate an ordering for the nodes to give a reduced bandwidth matrix. It should be noted however,

that this is not the optimum minimum bandwidth but merely a reduced bandwidth for any given set of connected nodes. The algorithm works in an advancing front method. Generation starts at a single point on the boundary; a new set is defined as those points immediately adjacent to the first set – each new set can be described as a *wave front*. These points are then added to the matrix in order according to the number of adjacent points they have. This continues until there are no more points left in the domain. This can be seen diagrammatically in the following example.

For the example domain given in Figure 8.2(a), consisting of 20 nodes, the matrix constructed from the node numbers can be seen to cover the matrix with plenty of empty space, Figure 8.2 (b). However, using the Cuthill-McKee algorithm, all the elements are arranged near to the diagonal. The nodes are numbered randomly to simulate the order in which the nodes would be generated using a Delaunay triangulation algorithm.

| **(a)** Example Node Structure | **(b)** Ordering Based on Node Numbering |
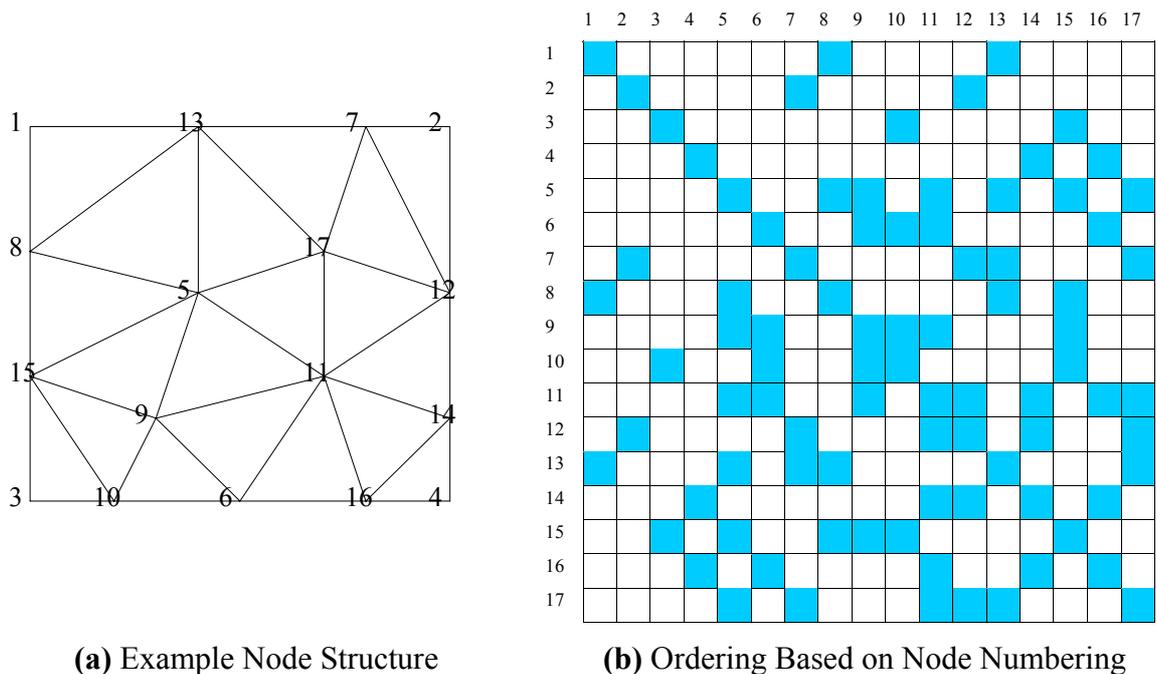
Figure 8.2*: Node Structure and Original Ordering*

The set of points belonging to each wave front are sorted according to their number of neighbours and then added to the matrix list in that order. The wave fronts and the order that they are generated can be seen in Figure 8.3:
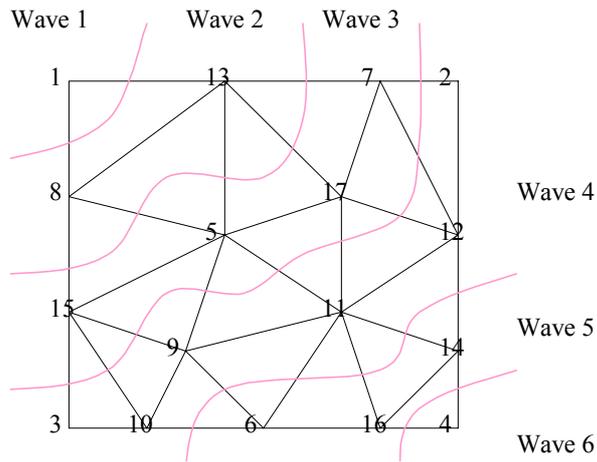
Figure 8.3: *Wave front Ordering*

The new ordering becomes: 1, 8, 13, 7, 15, 17, 5, 2, 3, 10, 12, 9, 11, 6, 14, 16, and finally 4, to give the matrix shown in Figure 8.4:
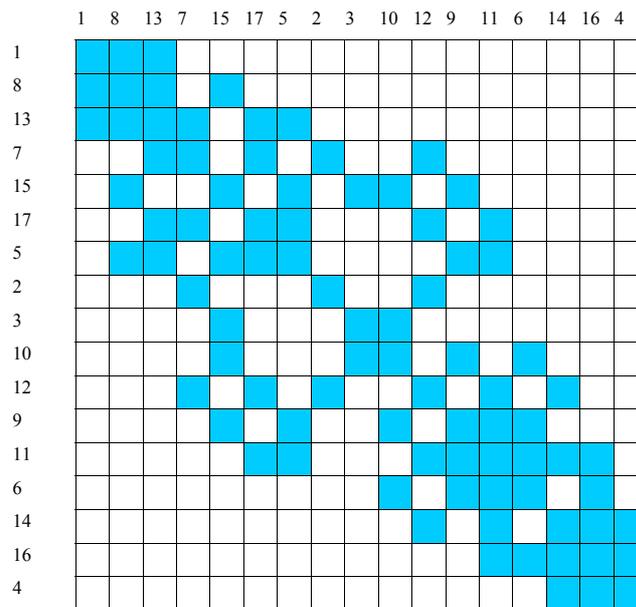
Figure 8.4: *Cuthill-McKee Ordering*

It can be seen from the above that there is a small reduction in the bandwidth; the following example, Figure 8.5, shows this more clearly.
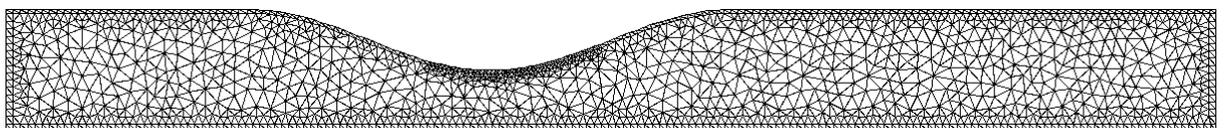
Figure 8.5: *Sample Mesh For a Symmetric Nozzle*

(a) Unordered Matrix　　　　　**(b)** Ordered Matrix
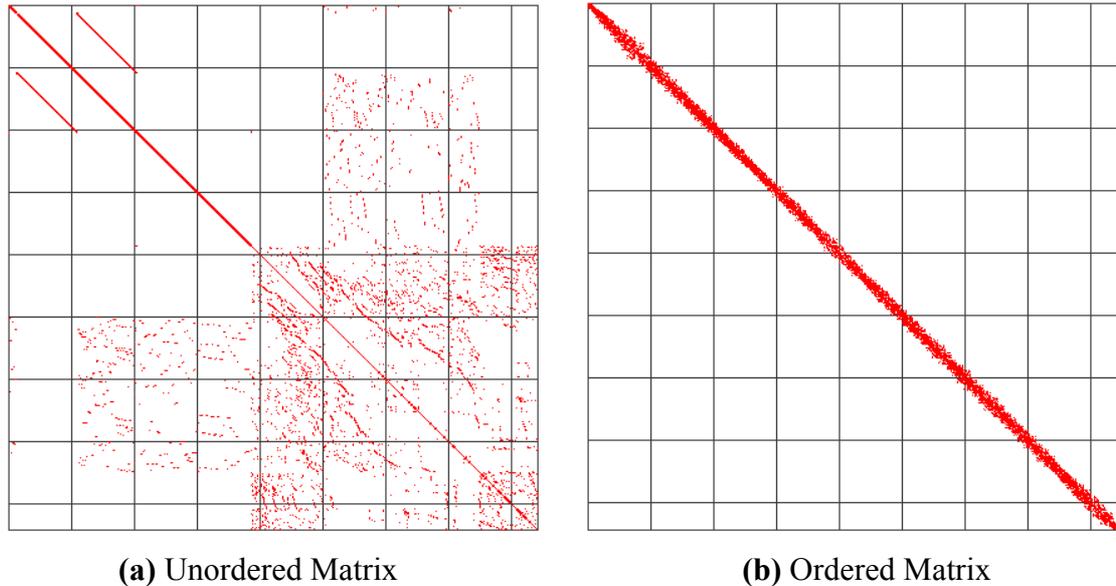
Figure 8.6*: Unordered and Cuthill-McKee Ordered Matrix*

The algorithm for the Cuthill-McKee method is as follows:

```
Mark all lists as empty - a NEW list, OLD list and MATRIX list
Add any node on the boundary to NEW list
Mark this point as visited, all other points as not visited
Append all points in NEW list to the end of MATRIX list

// MATRIX will hold the actual points in the order they
// will appear in the final matrix

Repeat {
    Clear OLD list
    Move all point in NEW list to OLD list
    Clear NEW list
    For all points in OLD list
    {
        For all neighbours of current point
        {
            If point has not already been visited, then add to NEW list
            Mark neighbour as visited
        }
    }
    Sort NEW list according to number of neighbours of each point
    Append all points in NEW list to the end of MATRIX list
} until all nodes have been checked.
```

Figure 8.7*: Cuthill-McKee Algorithm*

In some cases, it may still be necessary to decrease the memory requirements of the solver. A simple improvement, presented by the author, can be achieved by splitting the matrix into

sub-matrices and solving each of these independently. This effectively takes a section of nodes at a time, treating neighbours of this section as explicit. A disadvantage of this method is that the global matrix would need to be solved a number of times for information to propagate across the whole domain. In order to minimise information in those areas not covered by the sub-matrices, the sub-matrices are staggered on alternate iterations.
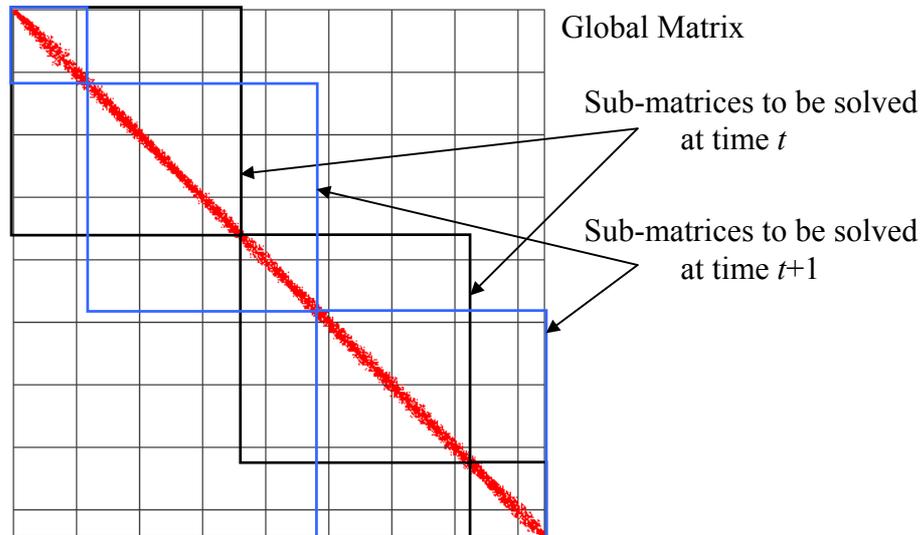
Figure 8.8: *Matrix Subdivision*

8.1   Concluding Remarks

An effective method of solving the system of equations has been presented in this chapter. Due to the limitations of memory and resources available, the matrix subdivision scheme shown above was necessary for the implicit scheme. This does not, however, lead to any detrimental effects. The bandwidth reduction was very effective in minimising the memory requirements for the preconditioner, and allowing subdivision of the global matrix.

9   ADAPTIVE GRID STRATEGIES

It has been shown in the previous chapters that a Riemann solver or equivalent can successfully capture the properties of a shock across two or three nodes. In the case of a coarse grid, the shock thickness will be many orders of magnitude smaller than the thickness of the shock defined on the grid. Actual shock waves will be in the range of 10^{-4}-10^{-5}m across depending on the local Mach number.

In order to correctly resolve this structure, it is necessary to define the grid such that the relevant number of nodes are in fact contained within this distance. Unfortunately, due to the nature of shock waves, *a priori* knowledge of their position and strength is usually unknown. It is therefore practical to define a method whereby the grid is refined only in those places where the property gradients exceed some previously defined limit.

No such adaptive strategies are applied to vortex shedding. As can be seen in the section on the Water-Table Model, the grid is refined in a block over the area in which the vortex shedding occurs.

The adaptive grid strategy is based on the steady state solution; a triangle is simply subdivided if its density gradient is above a user-defined limit. If the adapt limit was set to too low a value, density gradients associated with minor events such as vortex shedding would trigger adaption. As the process of vortex shedding is unsteady, with vortices convecting downstream, this would result in a large region of flow being adapted from the trailing edge to the outlet boundary. As this region can be determined beforehand, it is more efficient to generate a pre-defined adapted mesh here. Analysis of the experimental data has shown that the shock waves are present and do not move sufficiently from the steady case to warrant further adaptive steps. Therefore in the water table model, adaption is not required for unsteady flow.

In the current study, the grid is constructed of a triangular mesh. The gradient within each triangle is calculated using Greens' theorem, (9.1), based on the triangle in Figure 9.1:

$$\nabla \boldsymbol{Q} = \frac{1}{A} \sum_{f=1}^{3} \boldsymbol{Q}_f \hat{n}_f \qquad\qquad \textbf{(9.1)}$$
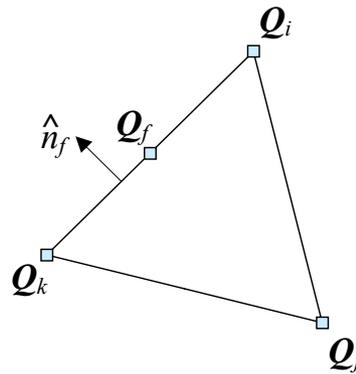
Figure 9.1*: Gradient in a Triangle*

Q_f is the property averaged at the midpoint of the face and $\hat{n}_f$ is the face normal multiplied be the length of that face.

9.1   Triangle Subdivision

If the gradient for any particular triangle exceeds the preset limits, the triangle is then subdivided as follows; new points are defined at the midpoints of the edges of the triangle. Together with the original three points, four non-overlapping triangles can be generated.

The two triangles formed by a bisection are linked in the subdivide routine and are known as 'split' triangles. Further subdivision requires that they are reformed into a single triangle and then subdivided into four as with a normal divide.

Triangle marked for subdivision
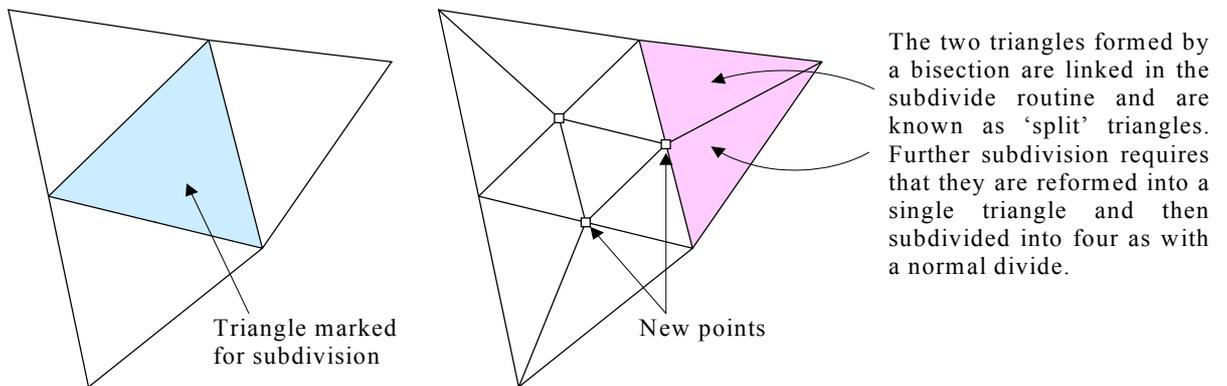
New points

Figure 9.2*: Triangle Subdivision*

It can also be seen from the above diagram that in order to accommodate changes to the central triangle, the adjacent triangles have been bisected to meet at the new points. In the event of two adjacent triangles required subdivision, they would simply meet at the midpoint of the common edge.

9.2 Boundary Subdivision

The boundary requires special consideration as it consists of two layers of points that must remain consistent. For this to hold, the author presents the following technique to ensure that both layers are subdivided equally - ensuring that the effective boundary does not move. This can be seen in Figure 9.3:
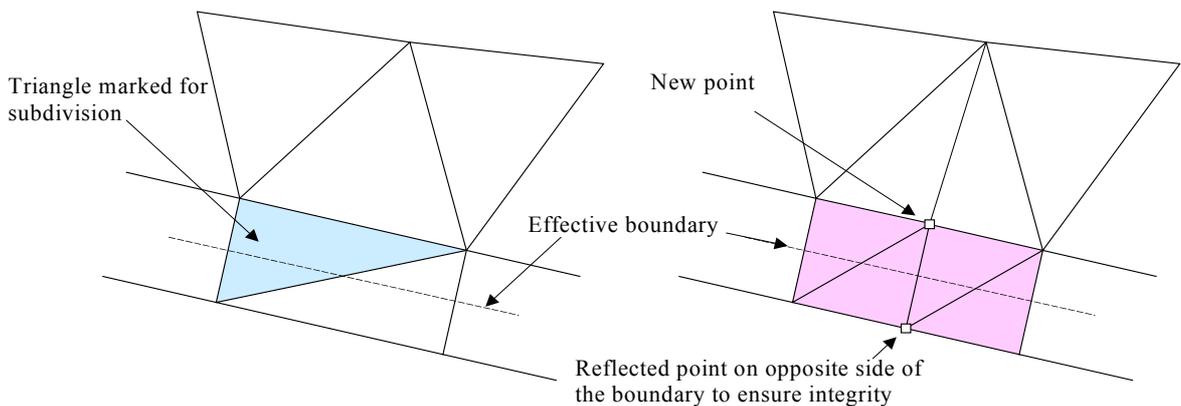
Figure 9.3: *Boundary Subdivision*

The basic algorithm for subdivision is given below.

```
For all Δs: {
     Calculate gradient within Δ
     If gradient > limit then {
          If normal Δ then mark for full-adapt
          If 'split' Δ then mark both it and the adjacent for full-adapt
     }
}


repeat {
     check all Δs: {
          if adjacent to a single marked Δ then mark for bisection
          if adjacent to 2 or 3 marked Δs then mark for full-adapt
     }
} until no more Δs need marking

For all Δs marked for full-adapt: {
     define new points at the midpoints of the edges
     split the edges in half
     define 3 new edges connecting the new points
     generate 4 Δs from the original and the new points
}
```

Figure 9.4: *Subdivide Algorithm*

9.3   Concluding Remarks

The adaptive strategy presented here is based on the ideas of Connell *et al* (1993) and a similar technique by Currie & Carscallen (1998). This method is also known as *red-green refinement*, Bank (1985). The algorithm is very complex, but equally versatile, and can also be used to adapt periodic boundaries in almost the same manner as a solid boundary. The results of this can be seen clearly in the adaption of the turbine blade geometry in chapter 11. The removal of points in regions of low density gradient has not been considered for this study.

10   NUMERICAL VALIDATION

In the following section, several numerical results are presented that test a class of problems associated with the Euler equations. Various two-dimensional geometries are presented, all being generated for the specific purpose of validating the developed code. The validation exercise is a fundamental stage in the development of a working code if numerical accuracy is to be achieved. Complex phenomena such as shock waves and contact discontinuities are present in the transonic turbine cascade under investigation and therefore form a large part of the test cases. Grid generation is highly flexible allowing the analysis of the complex geometries associated with the thesis, namely the turbine cascade and water-table geometries in the following sections. The validation exercises performed are:

- Flow over a supersonic compression corner
- Flow over a supersonic expansion corner
- Normal shock against a cylinder
- Subsonic flow through a converging-diverging nozzle
- Transonic flow through a converging-diverging nozzle
- Shock tube problem
- Supersonic flow over a backward facing step

10.1   Supersonic Compression Corner

The first test case is an oblique shock; this is a shock wave that is not perpendicular to the direction of the flow, and occurs when supersonic flow is deflected by a sharp object or by a wedge. A schematic of this can be seen in Figure 10.1.1 below:
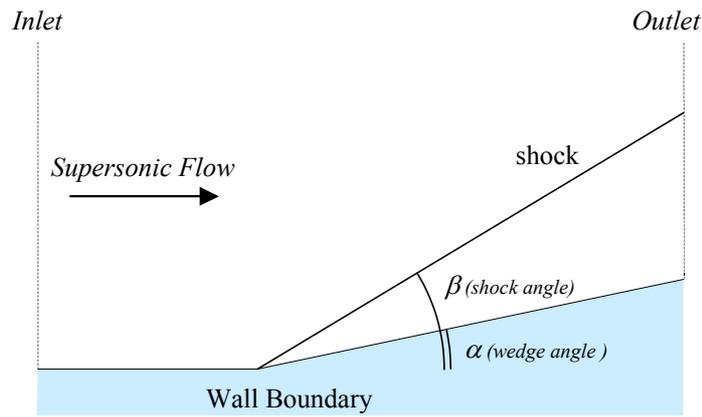
Figure 10.1.1: *Compression Corner*
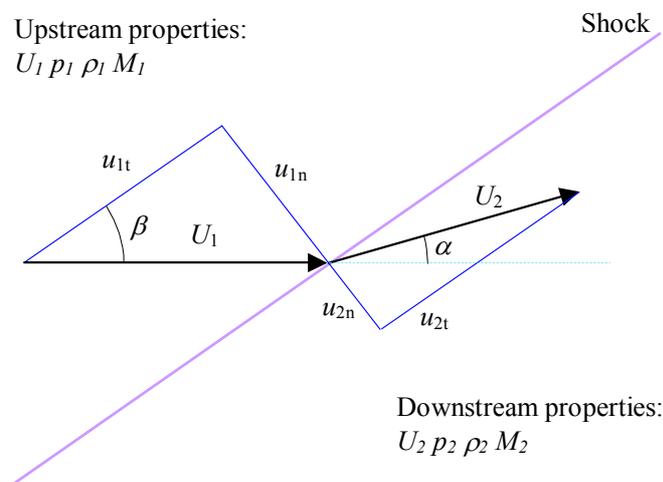
Figure 10.1.2: *Shock Wave as Normal and Tangential Components*

Supersonic inviscid flow over a compression corner has an exact solution that can be derived from the Rankine-Hugoniot equations (10.1.1) for an oblique shock against a wedge of a known angle α and a known supersonic inlet Mach number M_1. The flow is separated into normal and tangential components (Figure 10.1.2), the tangential component remains constant, and the normal component is treated as for a normal shock. Thus, only the normal velocity component is reduced, resulted in the flow deflection. The angle of the shock wave β is first computed from Rankine-Hugoniot equations:

$$\tan(\alpha) = \frac{M^2 \sin(2\beta) - 2\cot(\beta)}{M^2(\gamma + \cos(2\beta)) + 2}$$

$$\frac{\tan(\beta - \alpha)}{\tan(\beta)} = \frac{2 + (\gamma - 1)M^2 \sin^2(\beta)}{(\gamma + 1)M^2 \sin^2(\beta)}$$

(10.1.1)

The downstream flow properties can then be calculated from the following equations.

$$u_{1n} = U_1 \sin(\beta)$$
$$u_{1t} = U_1 \cos(\beta)$$
$$M_{1n} = M_1 \sin(\beta)$$
$$\vartheta = \frac{\tan(\beta - \alpha)}{\tan(\beta)}$$

(10.1.2)

$$u_{2t} = u_{1t}$$
$$u_{2n} = u_{1n}\vartheta$$
$$\rho_2 = \frac{\rho_1}{\vartheta}$$

(10.1.3)

$$p_2 = p_1 \left(\frac{2\gamma}{(\gamma + 1)} M_{1n}^2 - \frac{(\gamma - 1)}{(\gamma + 1)} \right)$$
$$M_2 = \sqrt{\frac{\left(u_{2n}^2 + u_{2t}^2\right)\rho_2}{\gamma p_2}}$$

(10.1.4)

For test case 1, the inlet conditions are: Mach number = 3.0, wedge angle $\alpha = 9.5°$, $p = 1.0$ and $\rho = 1.0$. The shock angle $\beta = 26.93°$ from (10.1.1), and from (10.1.2) the exact downstream conditions are:

$$\begin{bmatrix} M \\ u_n \\ u_t \\ \rho \\ p \end{bmatrix}_1 = \begin{bmatrix} 3.0 \\ 1.6077 \\ 3.1647 \\ 1.0 \\ 1.0 \end{bmatrix} \quad \text{therefore} \quad \begin{bmatrix} M \\ u_n \\ u_t \\ \rho \\ p \end{bmatrix}_2^{exact} = \begin{bmatrix} 2.5296 \\ 0.9936 \\ 3.1647 \\ 1.6180 \\ 1.9872 \end{bmatrix}$$

(10.1.5)

The five schemes under test are:

- First Order Scheme – using Roe's Riemann Solver
- Barth Scheme – Roe's Scheme with MUSCL Extrapolation
- Van Leer Flux Splitting – Second order flux splitting scheme with MUSCL extrapolation
- Quirk-minmod Scheme – Roe's Scheme with embedded TVD limiter
- Adapted Quirk-minmod Scheme – as above but with adaptive meshing.

The non-adapted and adapted meshes for the compression corner are shown in Figure 10.1.3 and Figure 10.1.4 respectively. Examination of the pressure contours in Figure 10.1.6 and Figure 10.1.7, shows that the shock in the first order scheme is smeared to about four times the width of the shock captured by the Quirk-minmod scheme. Figure 10.1.9 shows this to be about 20 nodes. The scheme is thus too highly diffusive and can be discounted from all further tests. The Barth scheme and the Van Leer scheme are almost indistinguishable, requiring 6 nodes across the shock. The Quirk-minmod scheme is slightly better requiring only 5 nodes. Downstream pressure values are highly accurate for all schemes. Examination of the density profile, Figure 10.1.10, and the Mach number profile, Figure 10.1.11, shows that all the non-adapted schemes do not recover the density or Mach number correctly across the shock. The Quirk-minmod scheme does however give the slightly better results. The results are acceptable once the scheme has been adapted though with an accompanied reduction in shock width. Figure 10.1.8 shows this more clearly when compared to Figure 10.1.7. In all cases, the solutions converge very quickly with the first order scheme being the quickest due to the high diffusivity. The higher order schemes all converge at about the same rate, Figure 10.1.12, with the adapted scheme taking more iterations due to the increased number of points within the mesh. All schemes do however converge to an acceptable residual level.

The downstream computed values at co-ordinates (5.0, 1.25) are:

$$
\begin{bmatrix} M \\ u_n \\ u_t \\ \rho \\ p \end{bmatrix}^{numerical}_{downstream} = \begin{bmatrix} 2.5266 \\ 0.9930 \\ 3.1626 \\ 1.6164 \\ 1.9874 \end{bmatrix} \tag{10.1.6}
$$

These can be seen to be very close to the exact values, with the maximum relative error being 0.12% for the Mach number.
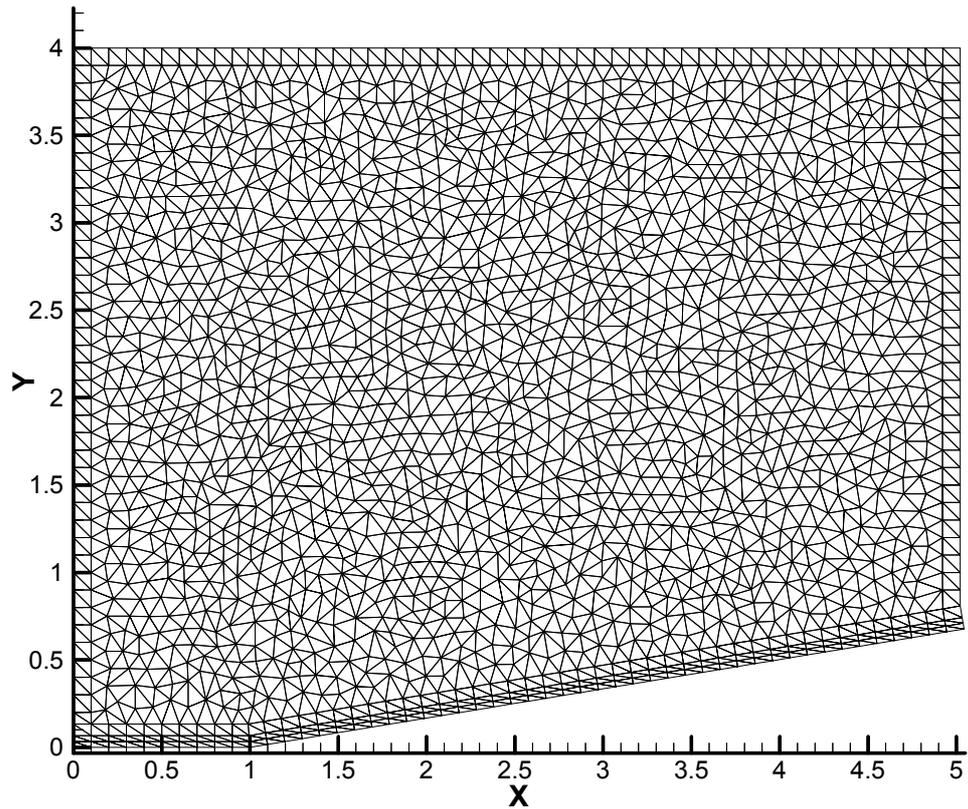
Figure 10.1.3*: Mesh for Compression Corner*
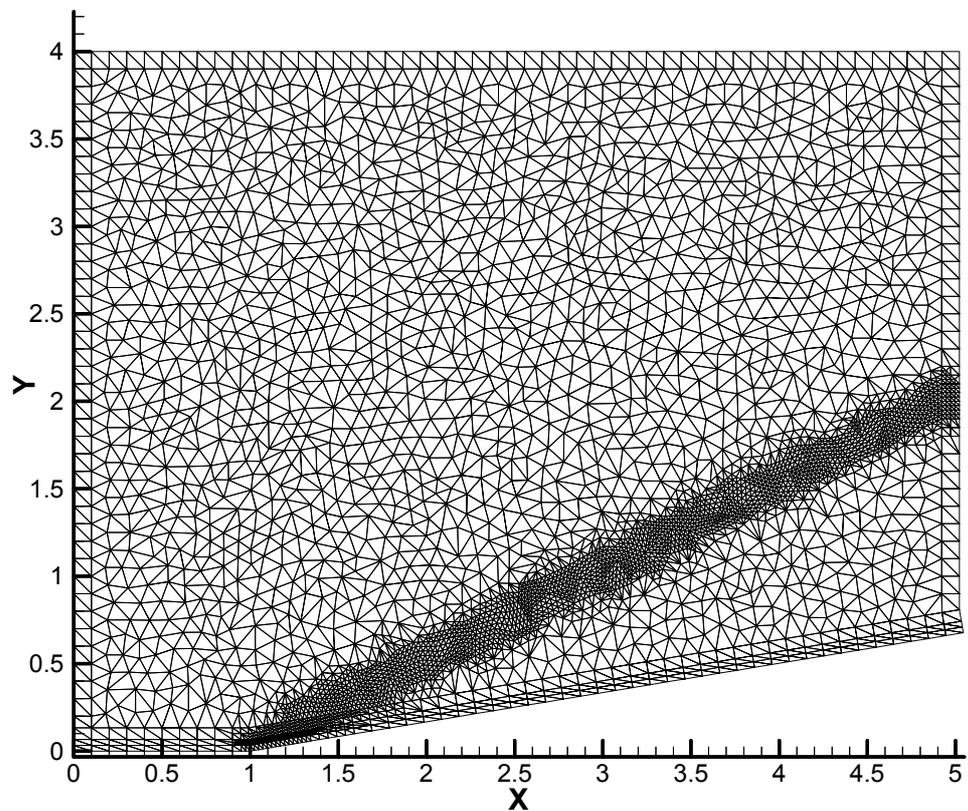
Figure 10.1.4*: Adapted Mesh for Compression Corner*
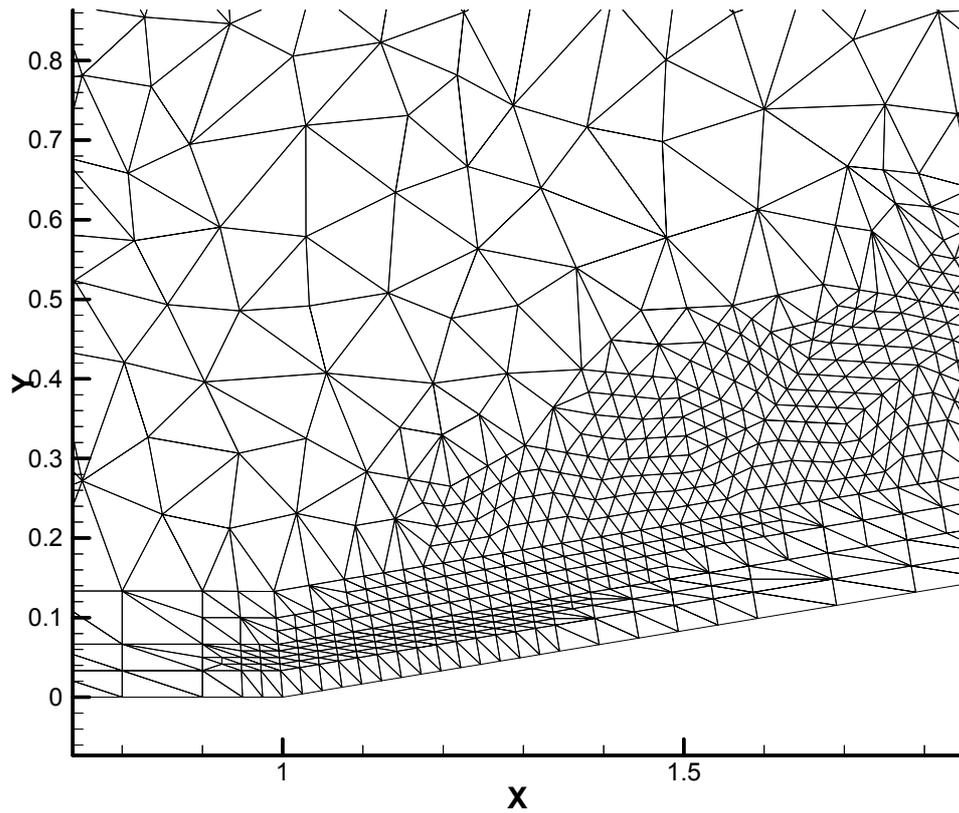
Figure 10.1.5*: Detailed View of Adapted Mesh for Compression Corner*
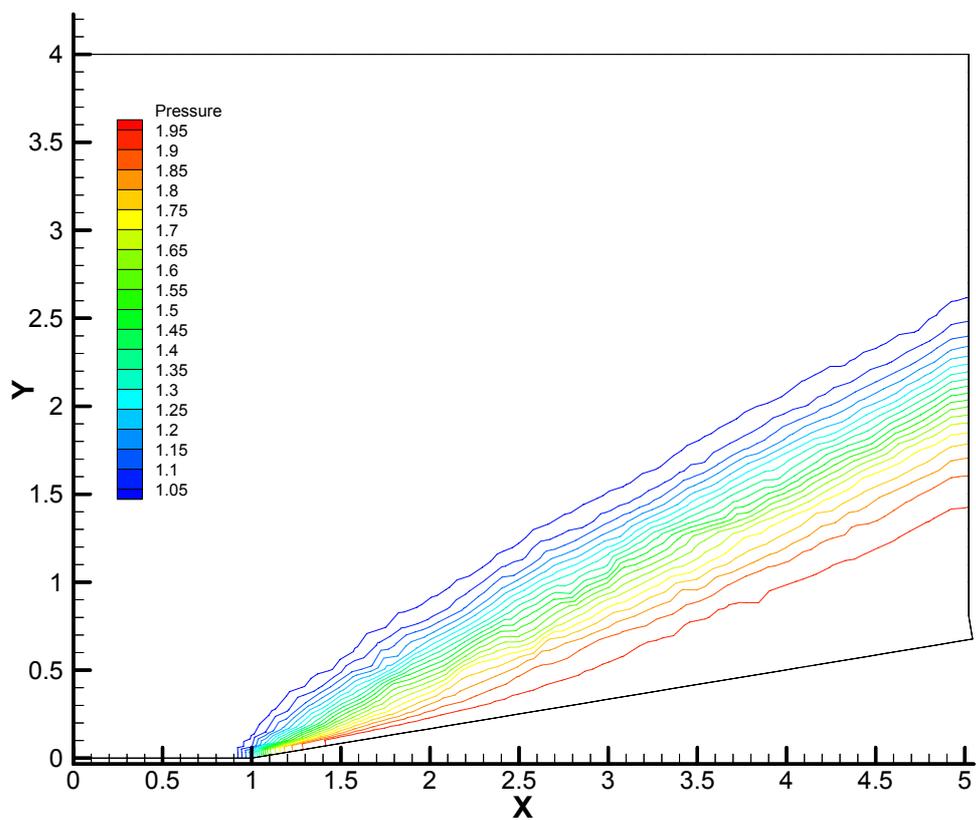
Figure 10.1.6: *Pressure Contours for First Order Scheme, Δp = 0.05*
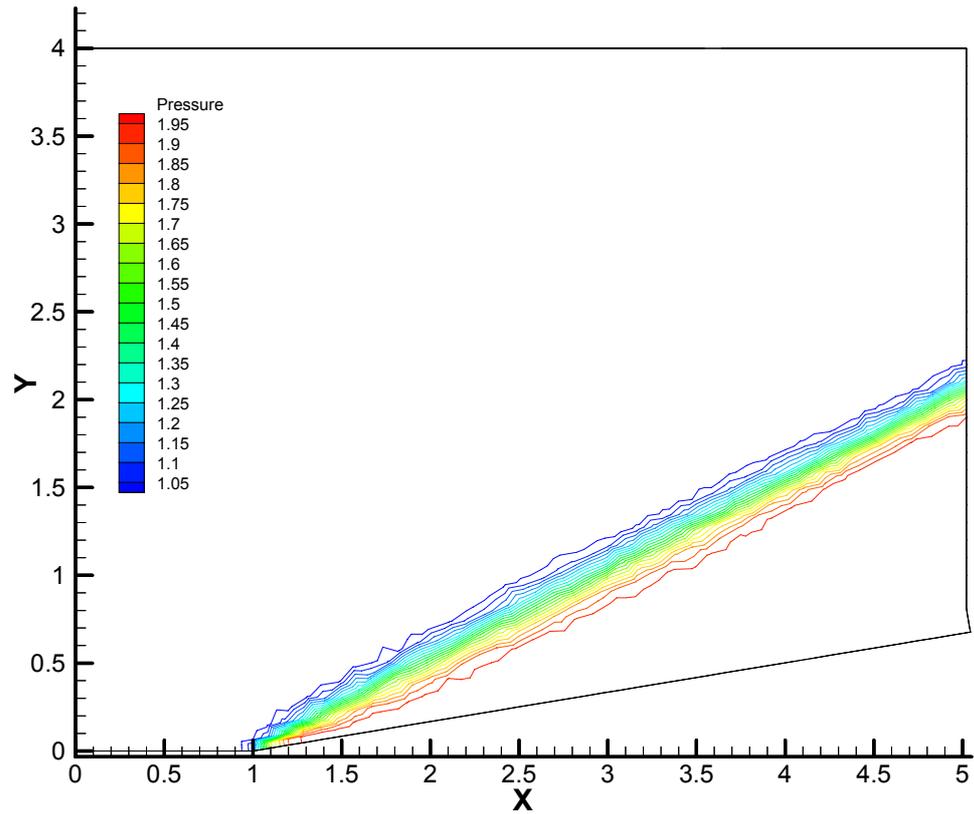
Figure 10.1.7: *Pressure Contours for Second Order Quirk-Minmod Scheme, Δp = 0.05*
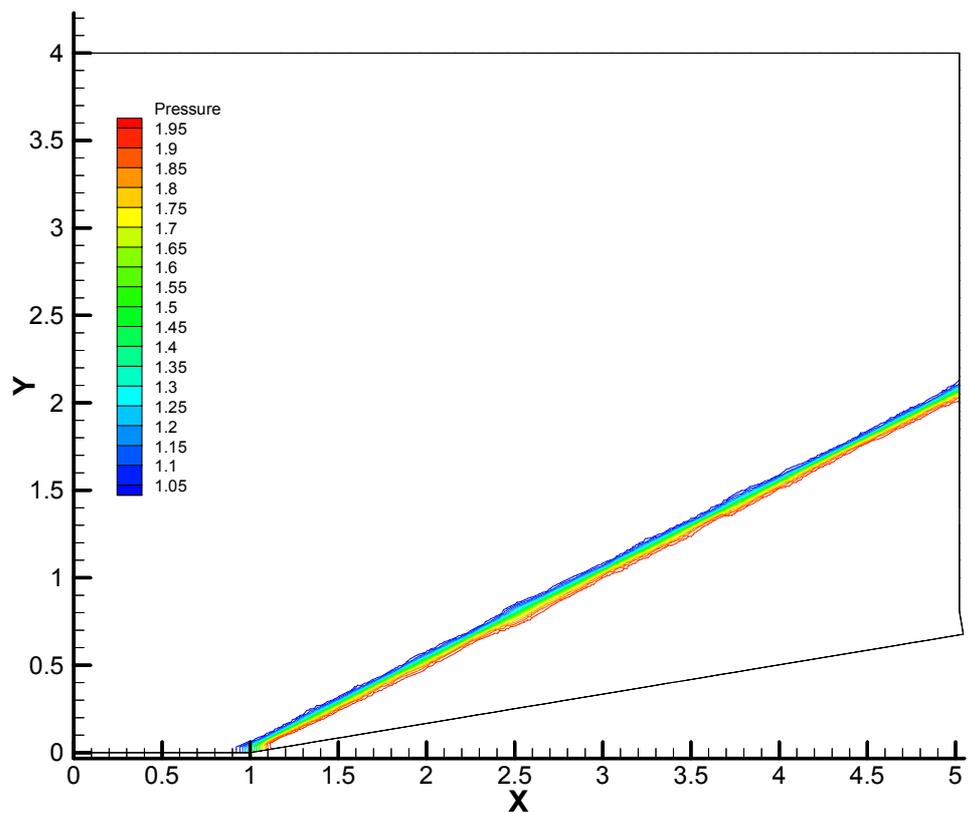
Figure 10.1.8: *Pressure Contours for Adapted Quirk-Minmod Scheme, Δp = 0.05*
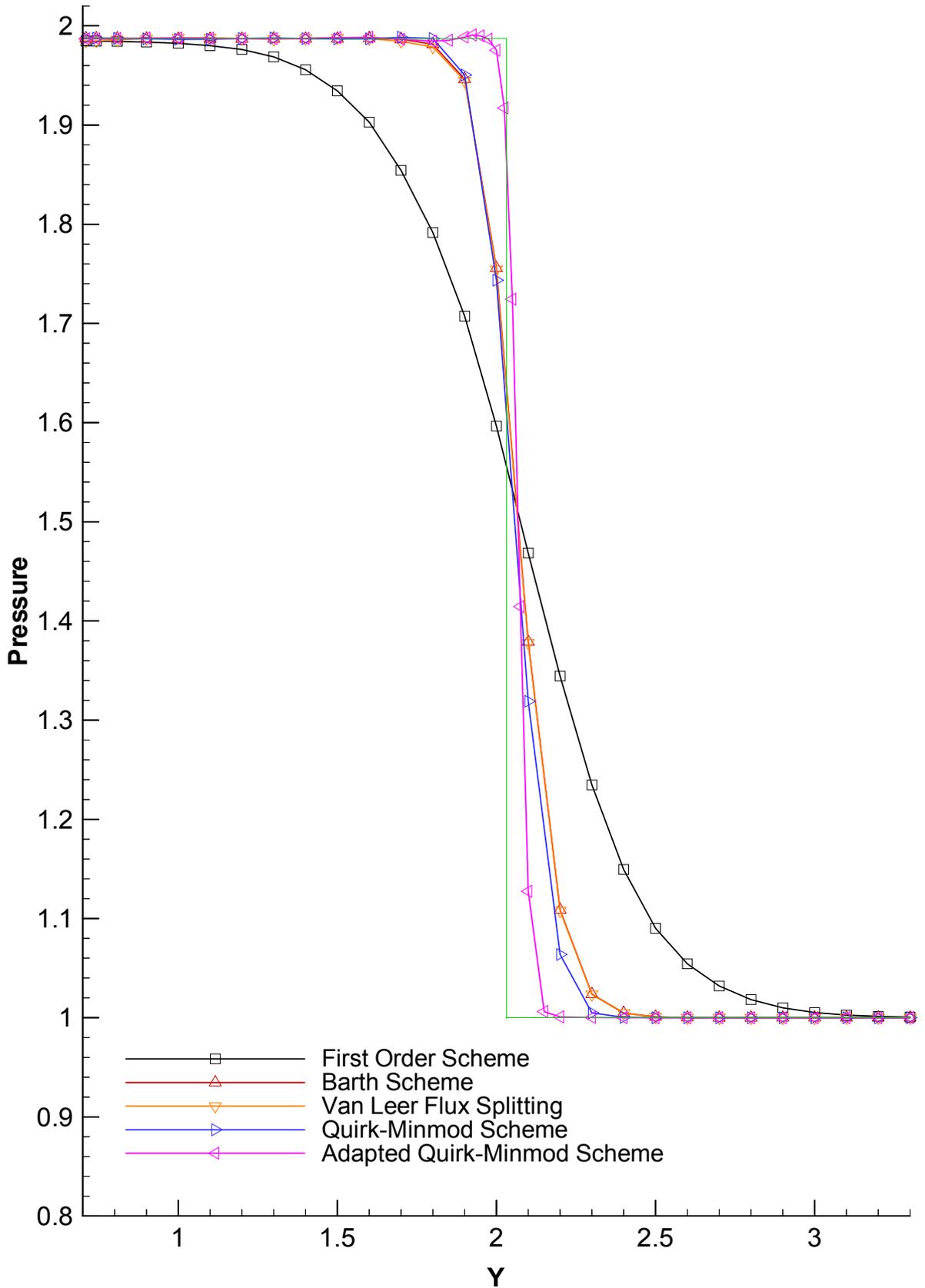
Figure 10.1.9: *Pressure Profiles for Exit Plane, x = 5.0*
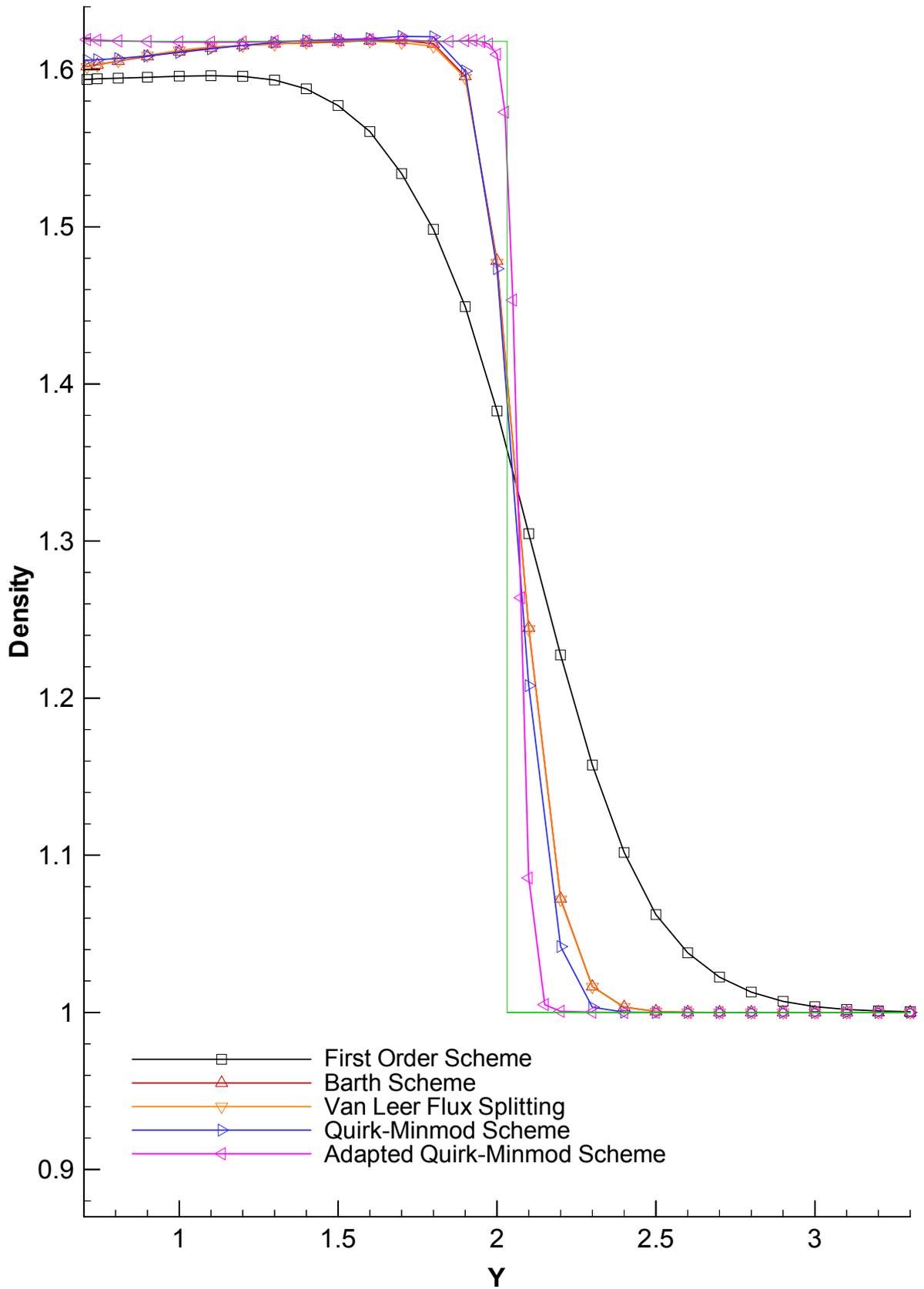
Figure 10.1.10: *Density Profiles for Exit Plane, x = 5.0*
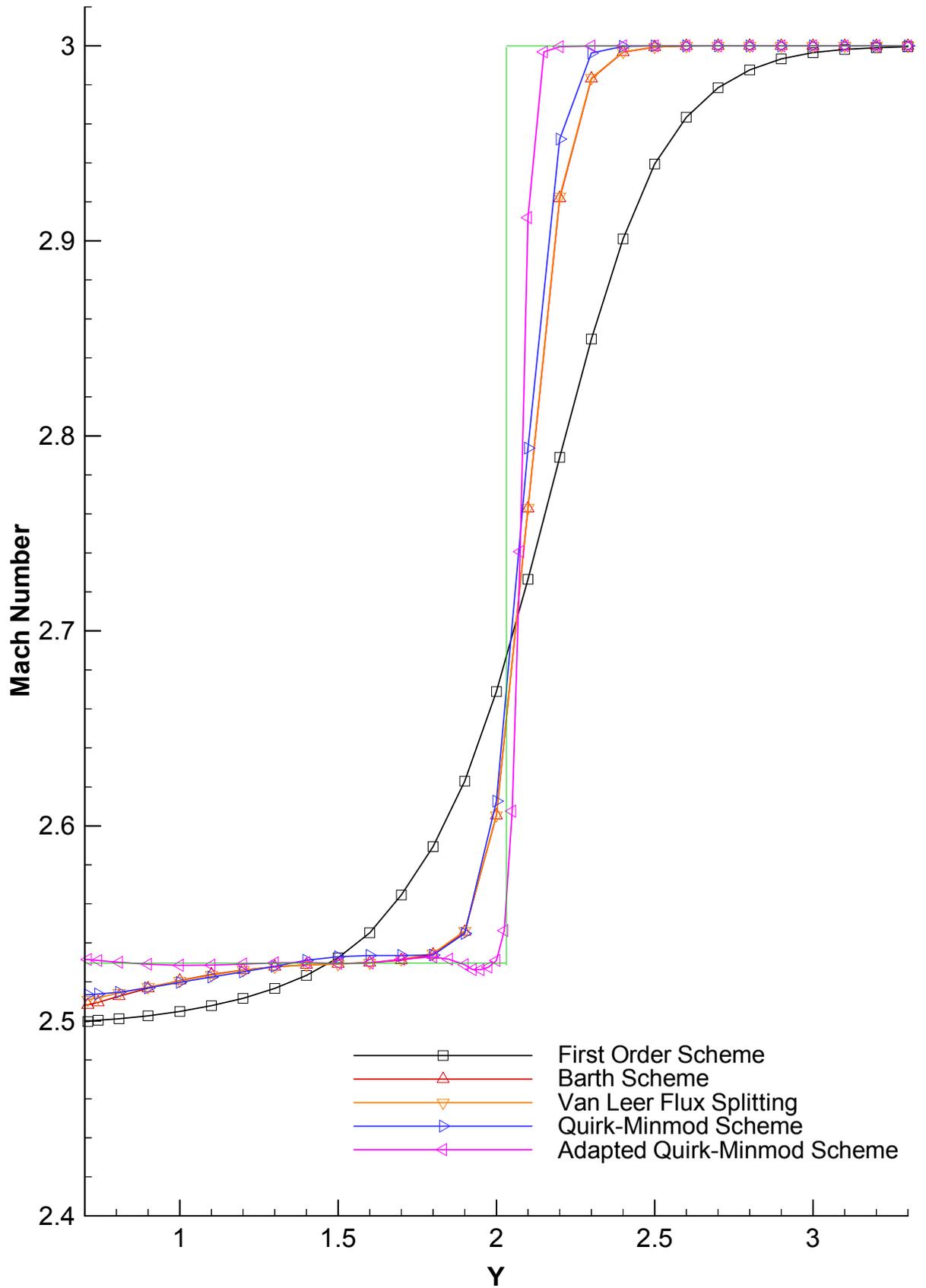
Figure 10.1.11: *Mach Number Profiles Across the Exit Plane, x = 5.0*
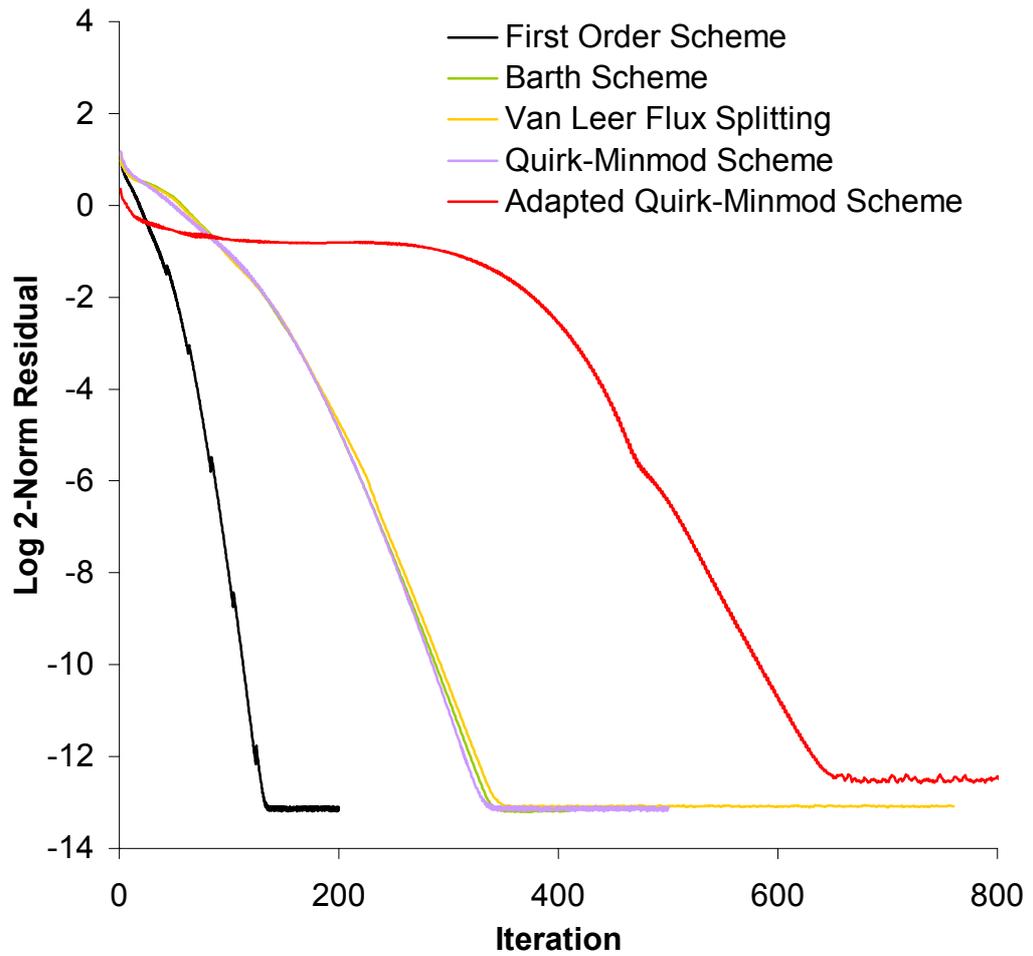
Figure 10.1.12: *Log 2-Norm Residual History for All Schemes*

10.2   Supersonic Expansion Corner

The second test case is supersonic flow over an expanding corner. This type of flow is known as a Prandtl-Meyer Expansion and is shown in Figure 10.2.1.
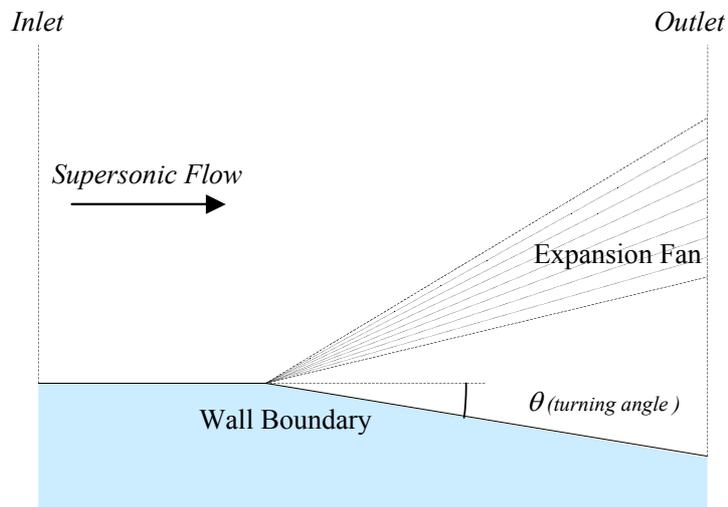
Figure 10.2.1*: Expansion Corner*
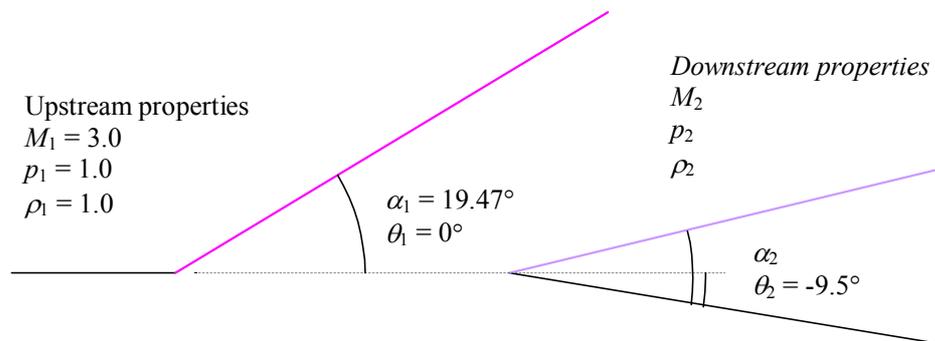
Figure 10.2.2: *Expansion Fan Conditions*

The effect of the turn in the wall is propagated along left running Mach lines in the characteristic plane. The exact solution is determined by a series of increments along the appropriate Mach line to give a continuous expansion. In practise, this can be calculated from isentropic and characteristic tables in a single step. For the current upstream conditions:

$$\begin{bmatrix} M \\ p \\ \rho \end{bmatrix}_1 = \begin{bmatrix} 3.0 \\ 1.0 \\ 1.0 \end{bmatrix} \tag{10.2.1}$$

From isentropic tables for $M = 3.0$:

$$\left(\frac{p}{p_0}\right)_1 = 0.02723 \tag{10.2.2}$$

$$\left(\frac{\rho}{\rho_0}\right)_1 = 0.07623 \tag{10.2.3}$$

Downstream conditions can be easily determining from the following rules:

$$\theta_2 - \theta_1 = -(\omega_2 - \omega_1) \qquad \text{for left running Mach waves} \tag{10.2.4}$$

$$\theta_2 - \theta_1 = \omega_2 - \omega_1 \qquad \text{for right running Mach waves} \tag{10.2.5}$$

where θ is the flow angle and ω is the angle on the characteristic plane.

The inlet flow angle $\theta_1 = 0°$, and from characteristic tables for $M_1 = 3.0$, $\omega_1 = 49.757°$, and $\alpha_1 = 19.47°$ (angle of the leading edge of the expansion fan).

For a turning angle of 9.5°, this implies that $\theta_2 = -9.5°$, therefore (10.2.4) gives $\omega_2 = 59.257°$.

Again, from characteristic tables, cross-referencing for ω_2 gives:

$$M_2 = 3.5461 \qquad \text{and} \qquad \alpha_2 = 16.38° \tag{10.2.6}$$

From isentropic tables for M_2:

$$\left(\frac{p}{p_0}\right)_2 = 0.01229 \tag{10.2.7}$$

$$\left(\frac{\rho}{\rho_0}\right)_2 = 0.04318 \tag{10.2.8}$$

Finally, the downstream pressure and density can be calculated:

$$p_2 = p_1 \frac{\left(\dfrac{p}{p_0}\right)_2}{\left(\dfrac{p}{p_0}\right)_1} = \frac{0.01229}{0.02723} = 0.4513 \tag{10.2.9}$$

$$\rho_2 = \rho_1 \frac{\left(\dfrac{\rho}{\rho_0}\right)_2}{\left(\dfrac{\rho}{\rho_0}\right)_1} = \frac{0.04318}{0.07623} = 0.5664 \qquad\qquad (10.2.10)$$

Therefore the exact downstream conditions are:

$$\begin{bmatrix} M \\ p \\ \rho \\ u \\ v \end{bmatrix}_2^{exact} = \begin{bmatrix} 3.5460 \\ 0.4513 \\ 0.5664 \\ 3.6940 \\ -0.6182 \end{bmatrix} \qquad\qquad (10.2.11)$$

The mesh for the expansion corner is given in Figure 10.2.3 for the non-adapted case, and Figure 10.2.4 for the adapted case. The schemes tested are those from the compression corner, less the first order case. Analysis of the pressure contours again shows that the adapted mesh, Figure 10.2.5, gives superior performance to the non-adapted mesh, Figure 10.2.6. Closer examination of the profiles shows that the Barth scheme gives excellent results for downstream pressure, shown in Figure 10.2.7, being close to the adapted case. However, the Mach number profile, Figure 10.2.9, shows only average performance with the Quirk-minmod scheme giving the better results. As with the compression corner, grid-adaption improves the downstream conditions. Taking a cross-section at the exit plane, Figure 10.2.10, it can be seen that the Mach number has now returned to the exact values calculated above, and is in accordance with the exact profile. The residuals in Figure 10.2.11 behave in an almost identical manner to the compression corner due to the supersonic nature of the test.

The downstream numerical values at co-ordinate (5.0, -0.25) are:

$$\begin{bmatrix} M \\ p \\ \rho \\ u \\ v \end{bmatrix}_2^{numerical} = \begin{bmatrix} 3.5462 \\ 0.4511 \\ 0.5662 \\ 3.6934 \\ -0.6181 \end{bmatrix} \qquad\qquad (10.2.12)$$
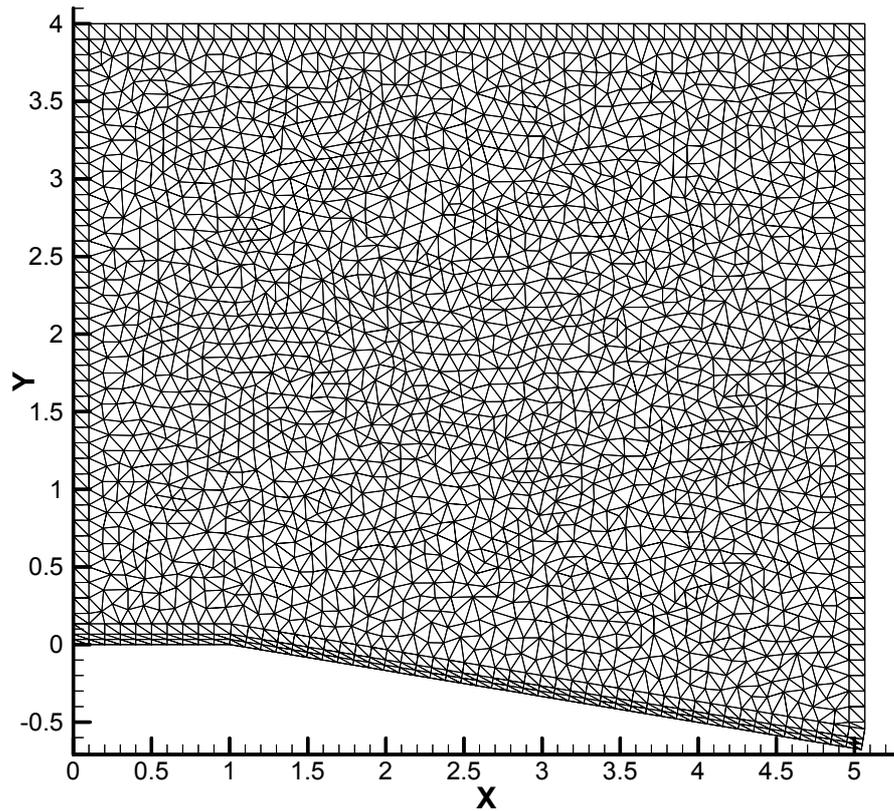
Figure 10.2.3: *Computational Mesh for Expansion Corner*
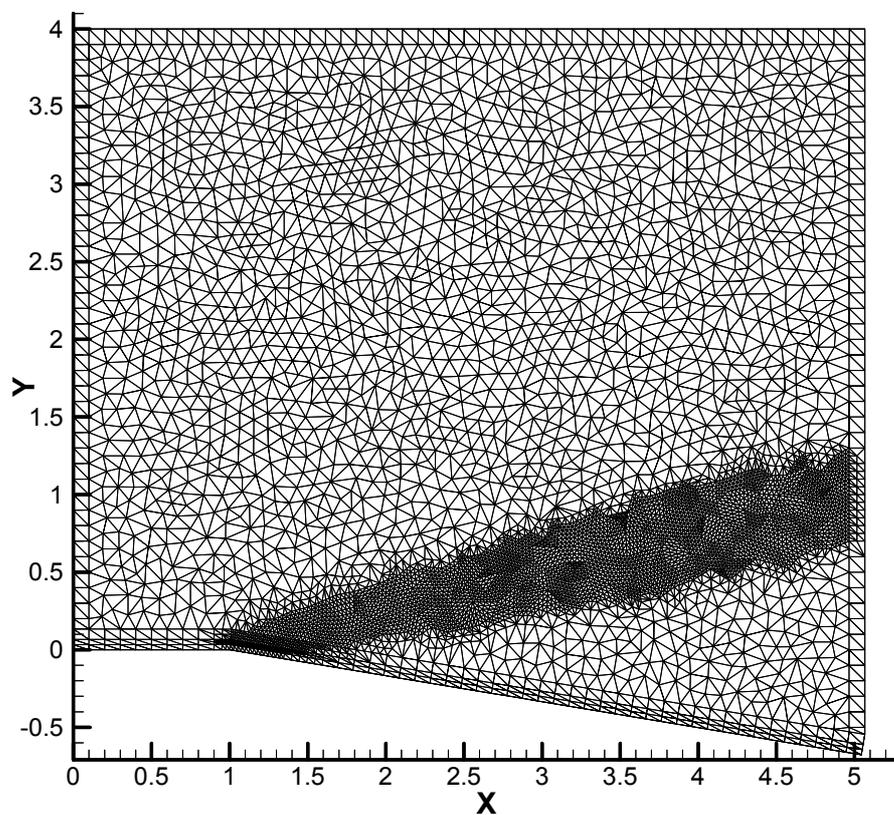
Figure 10.2.4: *Adapted Mesh for Expansion Corner*
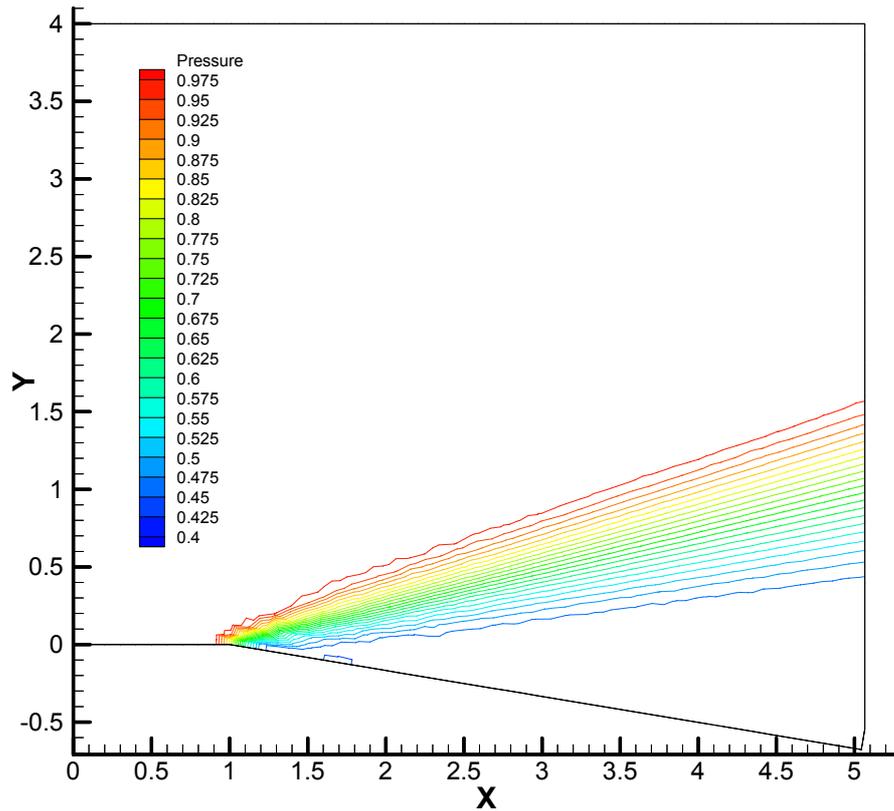
Figure 10.2.5: *Pressure Contours for Expansion Corner; Quirk-minmod Scheme, Δp = 0.025*
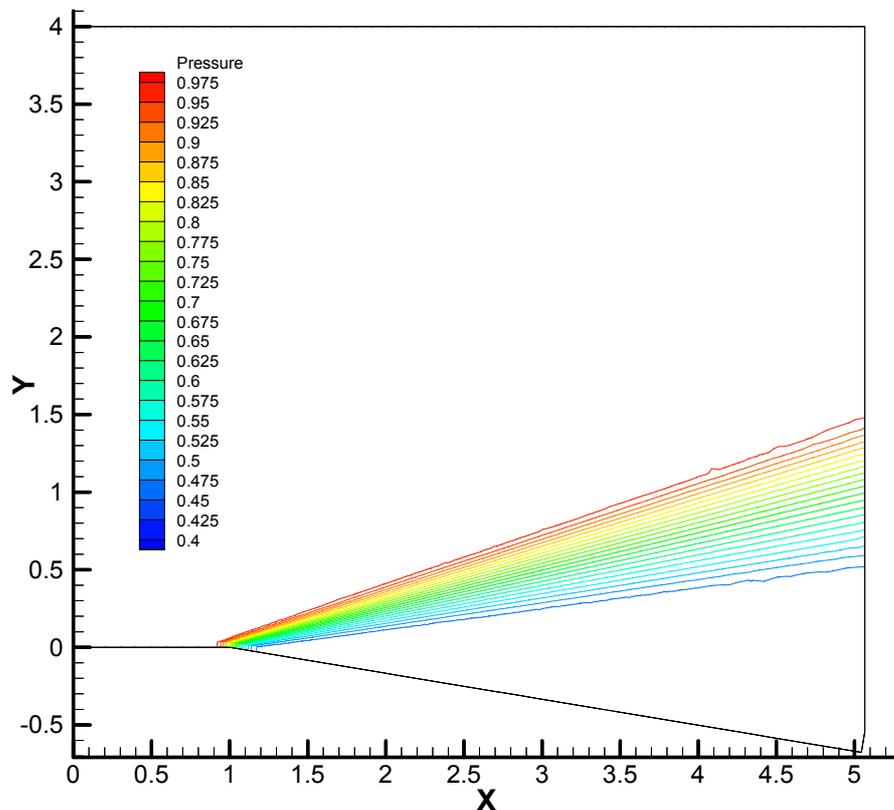
Figure 10.2.6: *Pressure Contours for Adapted Mesh; Quirk-minmod Scheme, Δp = 0.025*
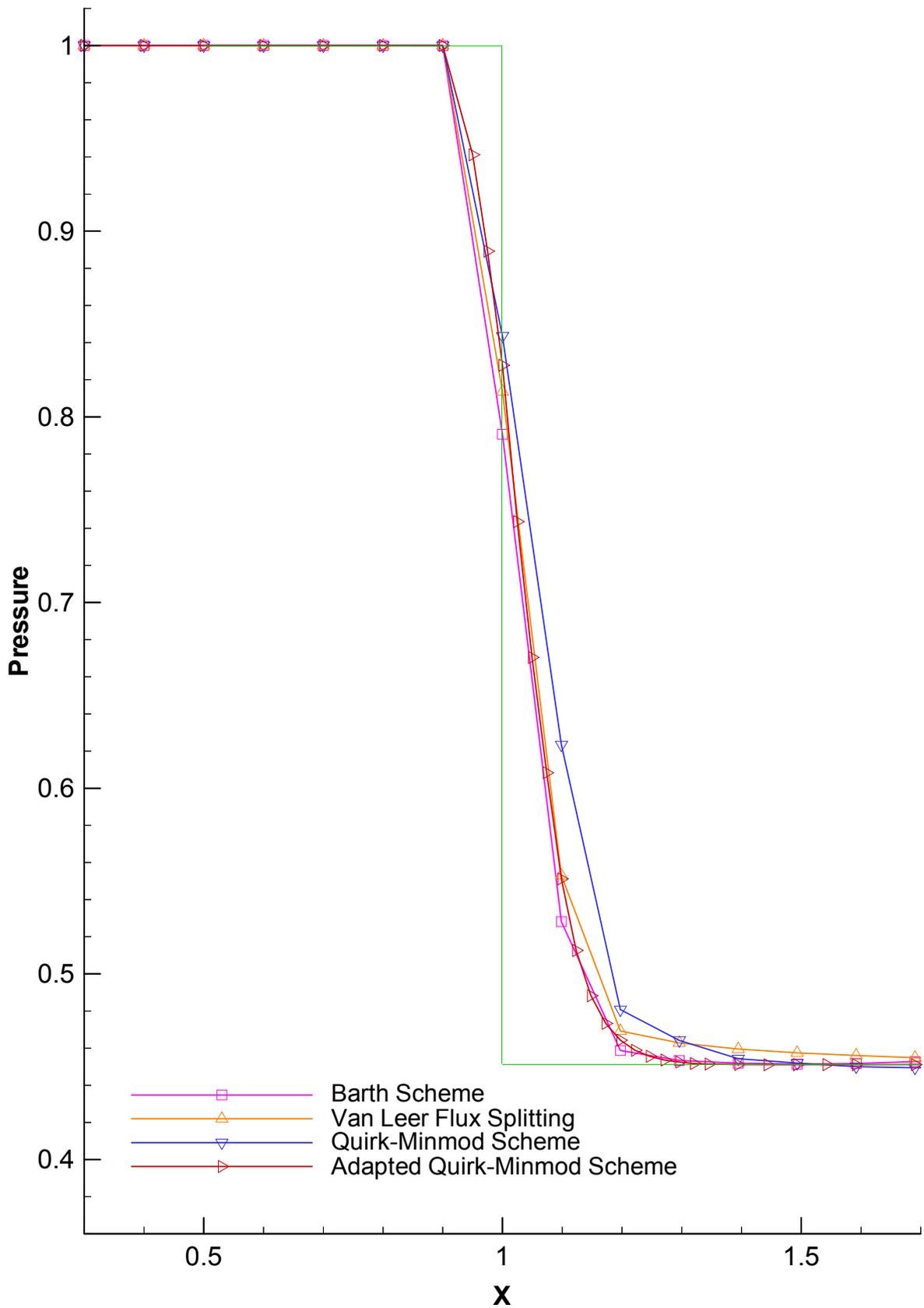
Figure 10.2.7: *Pressure Profiles Along the Lower Wall; Corner located at x = 1.0*
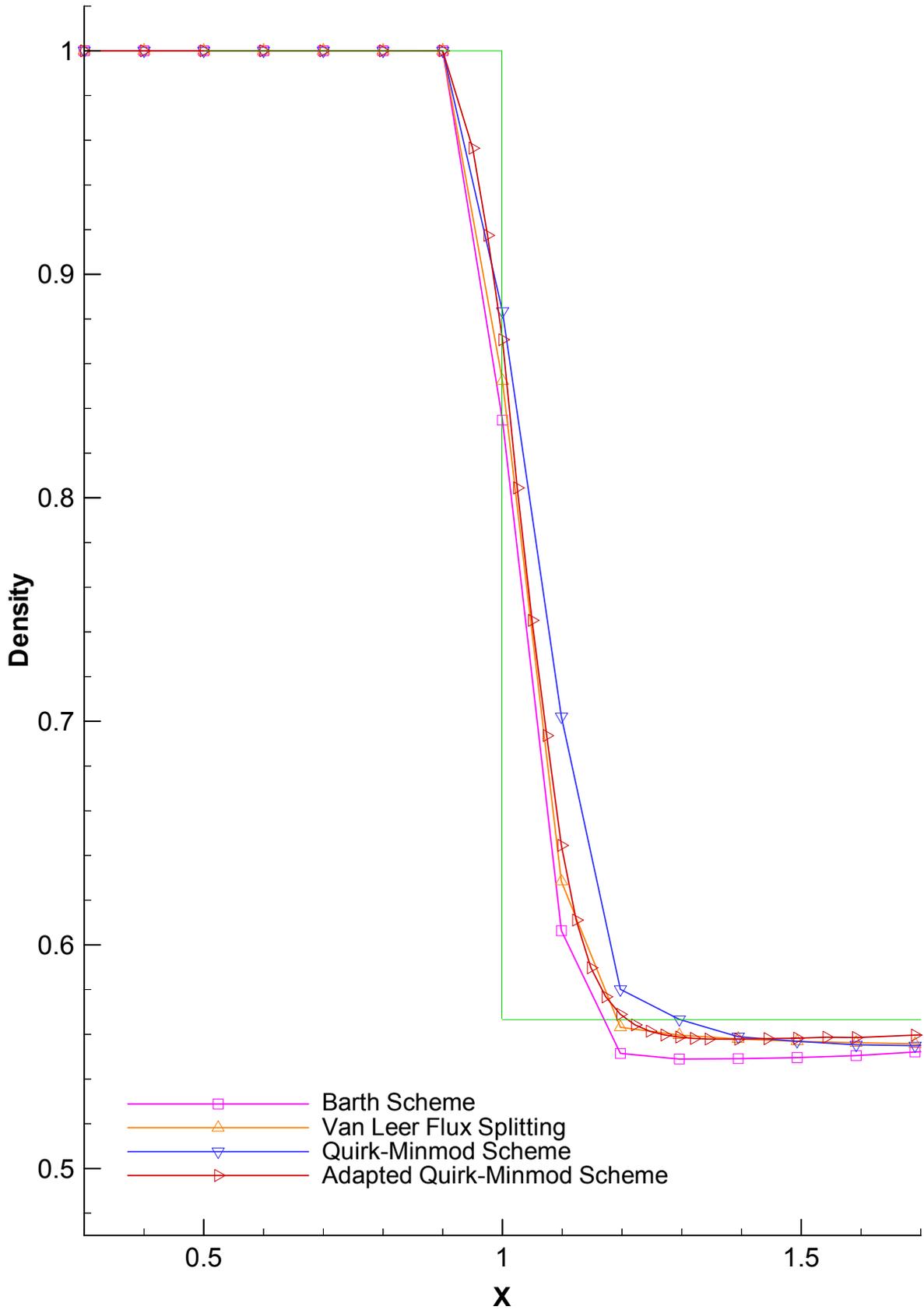
Figure 10.2.8: *Density Profiles Along the Lower Wall; Corner located at x = 1.0*
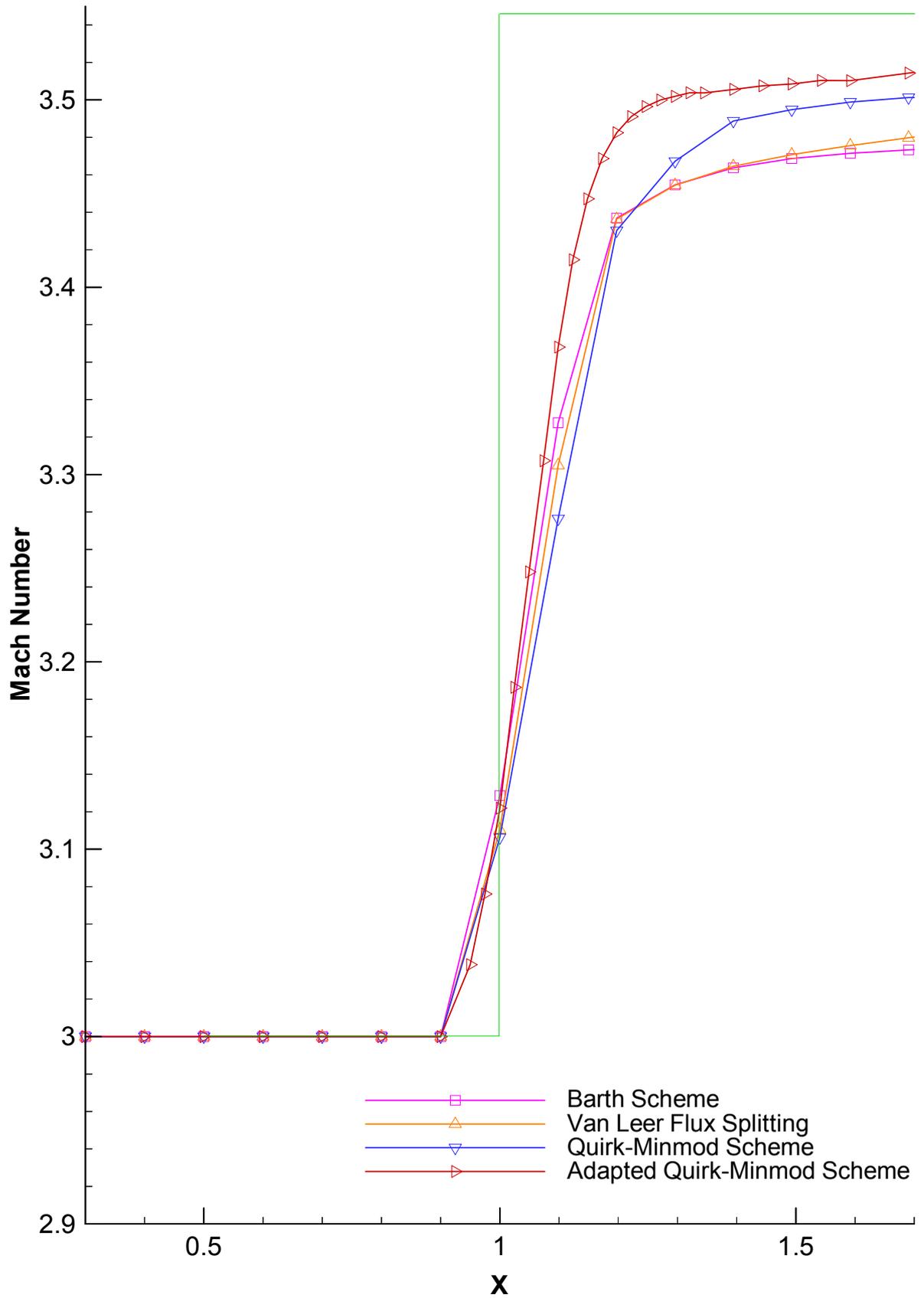
Figure 10.2.9: *Mach Number Profiles Along the Lower Wall; Corner located at x = 1.0*
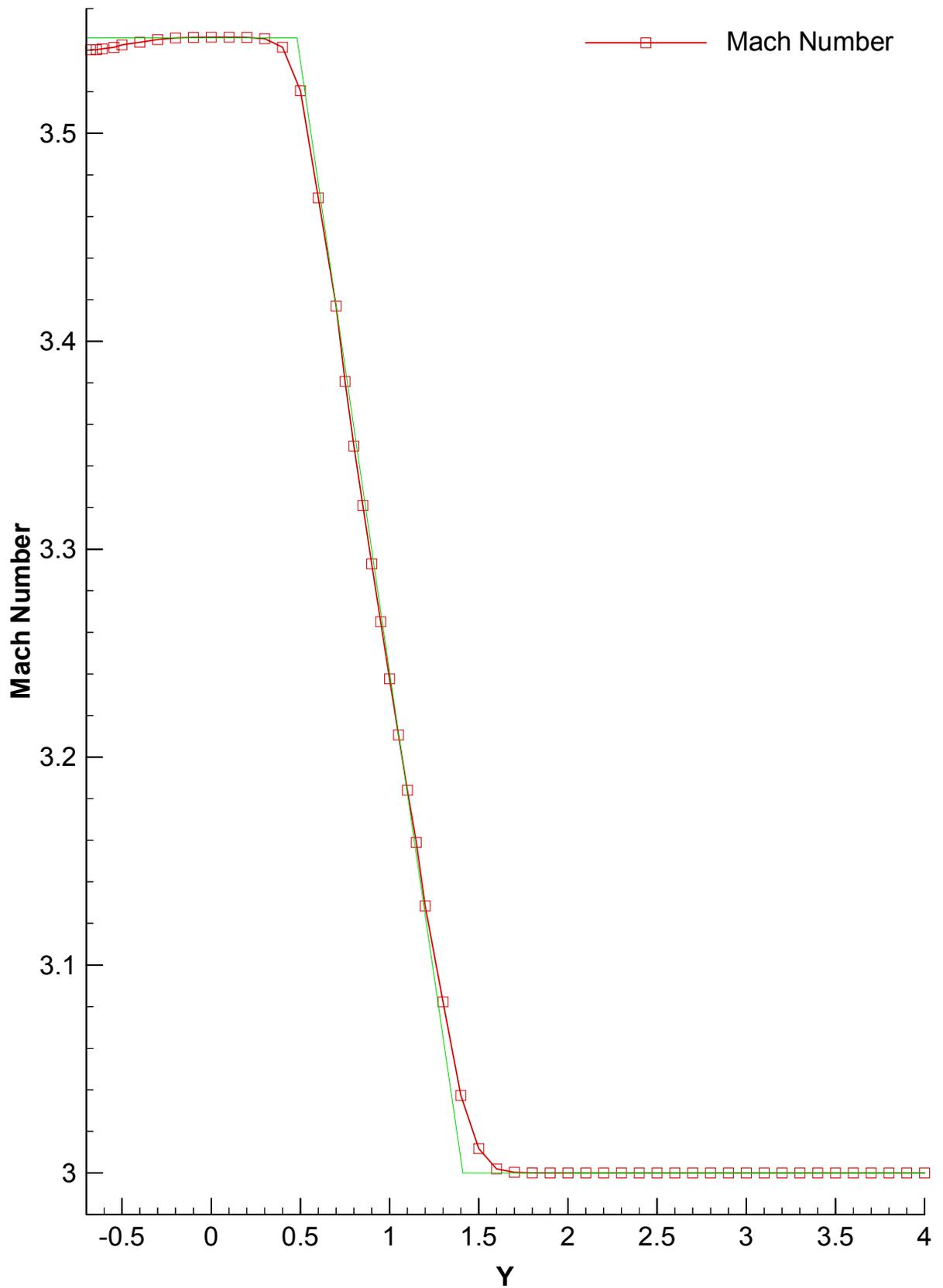
Figure 10.2.10: *Mach Number Profile Across the Exit Plane; Quirk-minmod Scheme, x = 5.0*
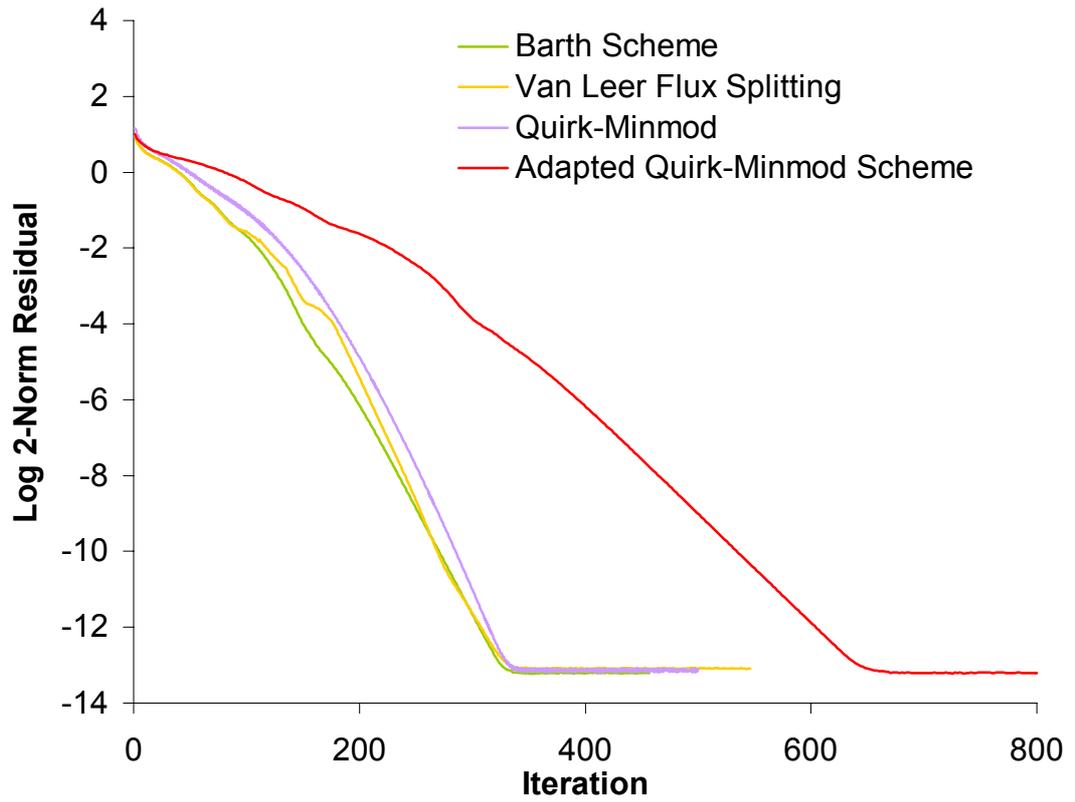
Figure 10.2.11: *Log 2-Norm Residual History for All Schemes*

10.3   Normal Shock Against a Cylinder

Test case 3 is supersonic flow impinging on a cylinder. This causes a normal shock to form against the leading edge. The performance of the solver can be compared against exact data extracted from normal shock tables. The geometry can be seen in Figure 10.3.1.
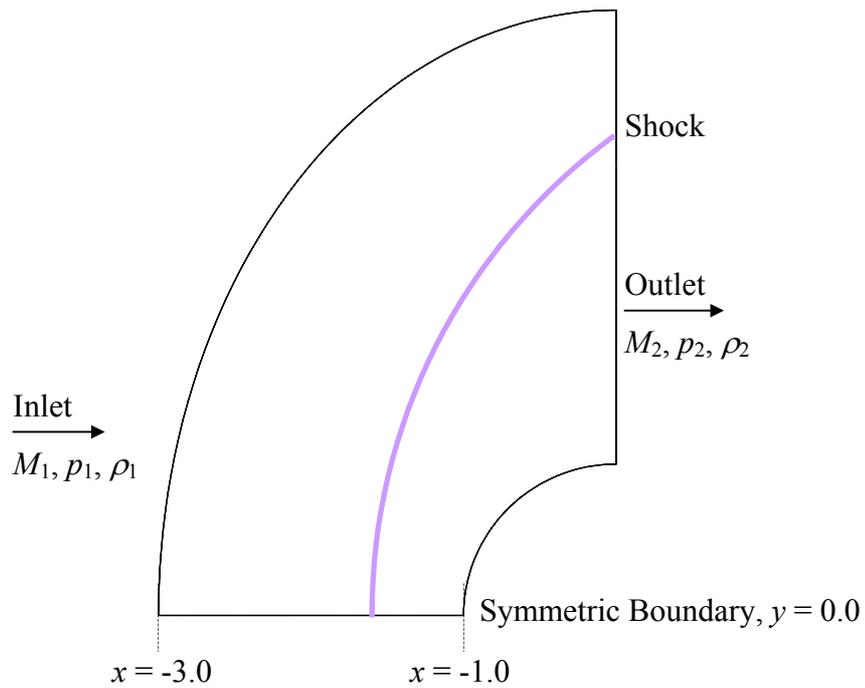
Figure 10.3.1: *Normal Shock Against a Cylinder*

The inlet conditions are defined by:

$$\begin{bmatrix} M \\ p \\ \rho \end{bmatrix}_1 = \begin{bmatrix} 3.0 \\ 1.0 \\ 1.0 \end{bmatrix} \tag{10.3.1}$$

From normal shock tables, the conditions immediately downstream of the shock are:

$$\begin{bmatrix} M \\ p \\ \rho \end{bmatrix}_2^{exact} = \begin{bmatrix} 0.47519 \\ 10.3333 \\ 3.8571 \end{bmatrix} \tag{10.3.2}$$

Behind the shock, the flow then stagnates against the cylinder. The flow properties along the centre-line, y = 0.0, from the shock to the wall can be calculated from the isentropic relations to give table 10.3.1:

x	*M*	*p*	ρ
-3.00000	3.0	1.0	1.0
-1.73733	3.0	1.0	1.0
-1.720	0.47519	10.3333	3.8571
-1.688	0.45000	10.49612	3.90044
-1.616	0.40000	10.80183	3.98125
-1.544	0.35000	11.08114	4.05452
-1.472	0.30000	11.33076	4.11955
-1.400	0.25000	11.54765	4.17955
-1.328	0.20000	11.72909	4.22248
-1.256	0.15000	11.87275	4.25936
-1.184	0.10000	11.97675	4.28598
-1.102	0.05000	12.0397	4.30206
-1.00000	0.00000	12.0608	4.30744

Table 10.3.1: *Flow Properties for Normal Shock*

The non-adapted and adapted meshes are shown in Figures 10.3.2 and 10.3.3 respectively. The higher performance of the adapted mesh can be seen in the pressure contours of Figure 10.3.5 compared to those in Figure 10.3.4. Pressure, density and Mach number profiles along the centerline, y = 0.0, are given in Figures 10.3.6, 10.3.7, and 10.3.8. The actual profiles are in good agreement with the exact values denoted by the green line. However, there are several issues of concern: undershoot and overshoot can be seen on either side of the shock in all profiles, implying the scheme is not performing as a TVD scheme should be, and although the Mach number is in excellent agreement this is at the expense of a drop in pressure and density at the wall. The author attributes this to the walls not functioning in a perfectly adiabatic manner. The residual log, Figure 10.3.9, is generally decreasing, although this does take a considerable number of iterations compared to the supersonic compression and expansion corners. The two predominant peaks at about iteration 2000 are the points at which adaption takes place.
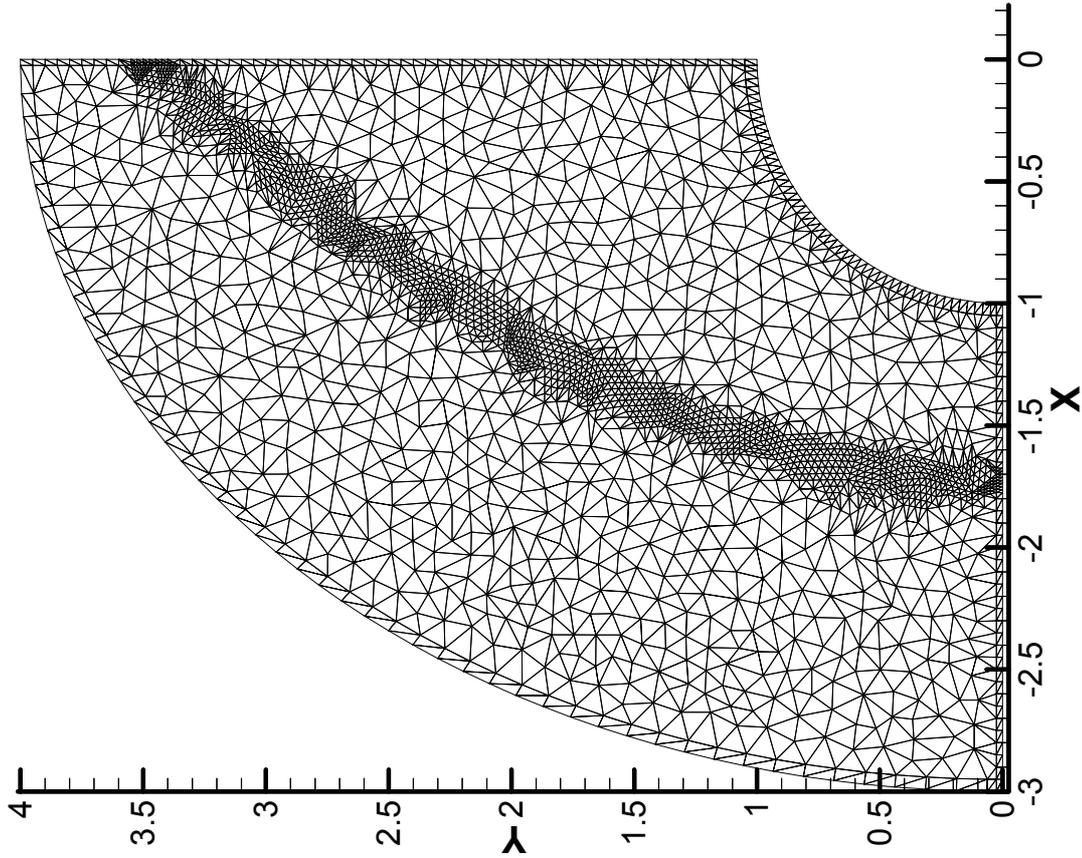
Figure 10.3.3: *Adapted Mesh for Quadrant, 2 Adaptive Steps*
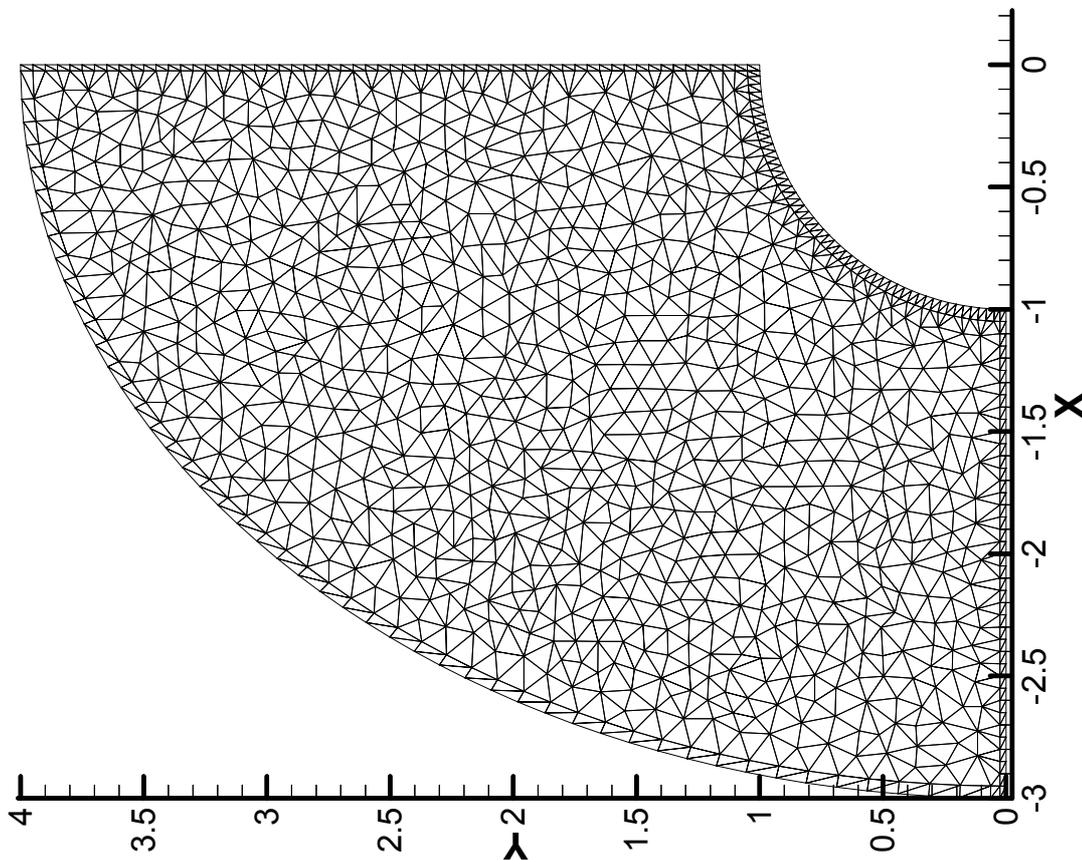
Figure 10.3.2: *Computational Mesh for Quadrant*

Figure 10.3.5: *Pressure Contours for Adapted Quadrant, Δp = 0.5*

Figure 10.3.4: *Pressure Contours for Quadrant, Δp = 0.5*
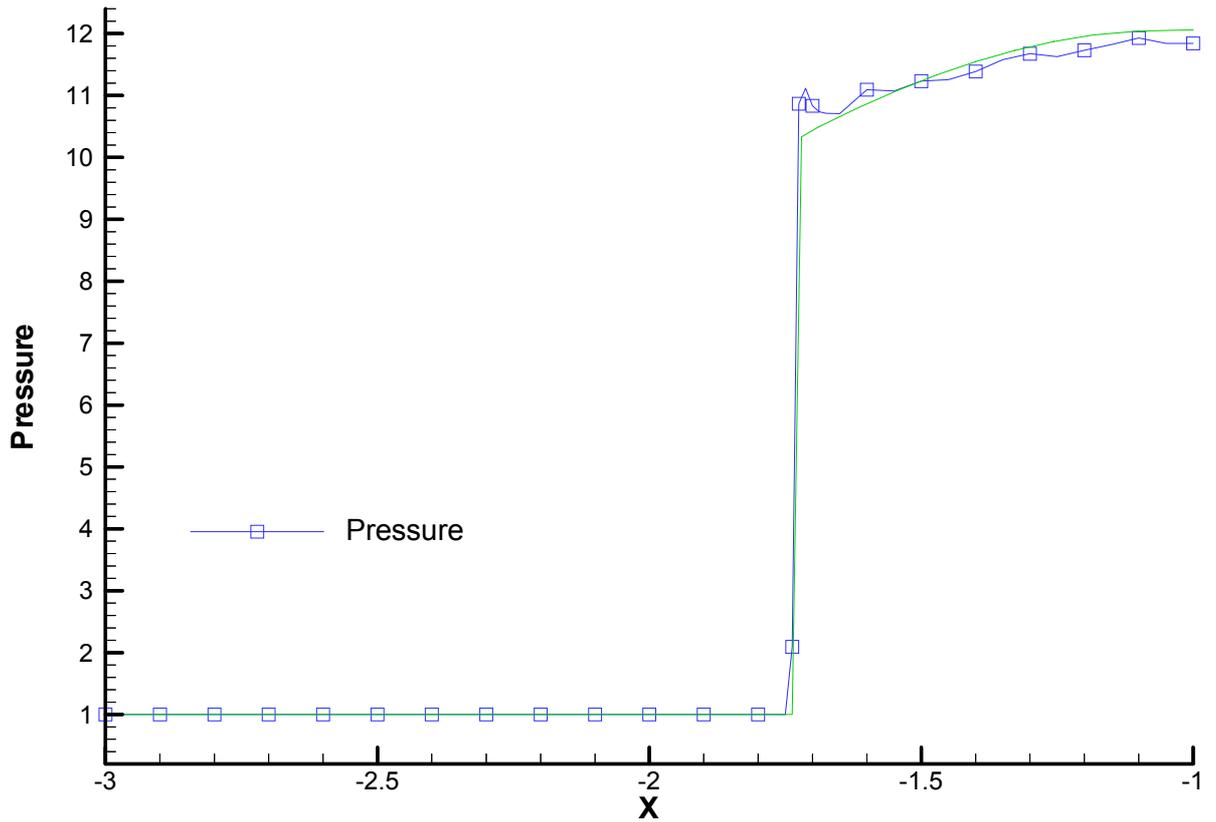
Figure 10.3.6: *Pressure Profile for Adapted Quadrant, y = 0.0*
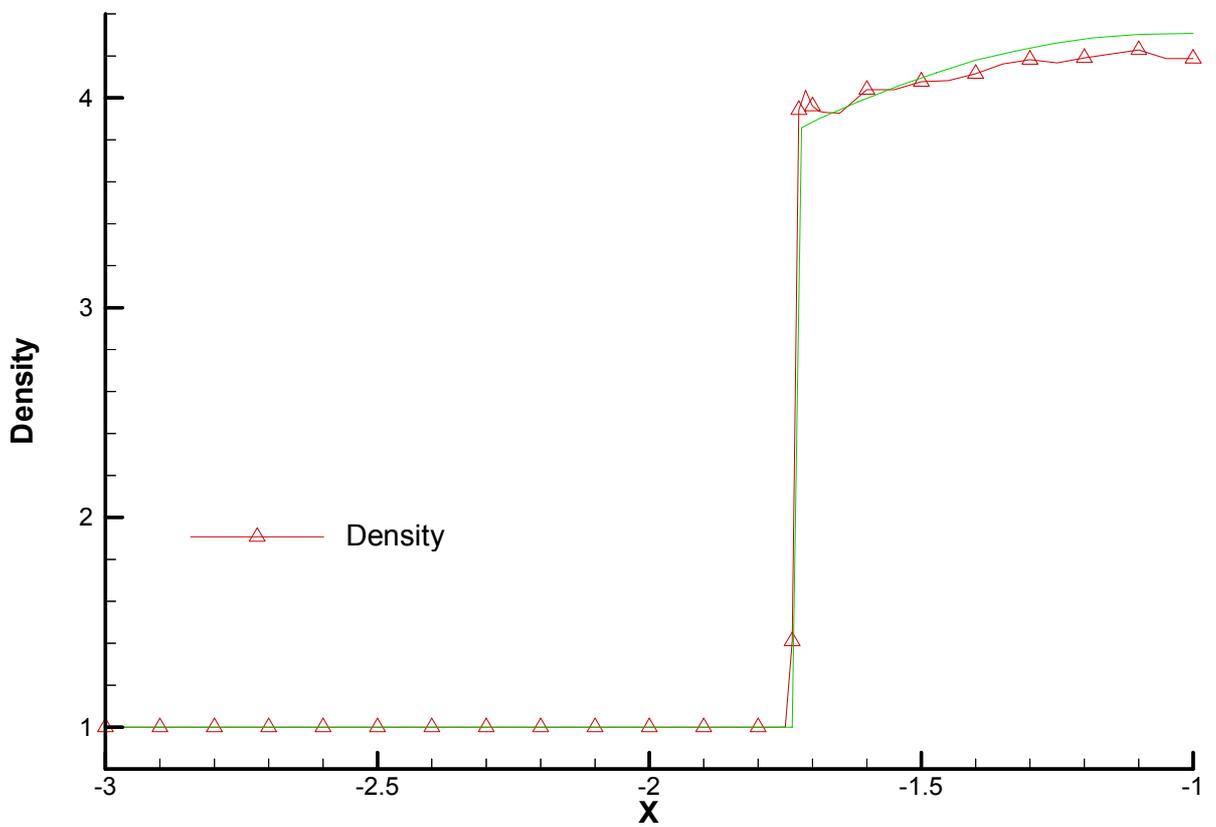
Figure 10.3.7: *Density Profile for Adapted Quadrant, y = 0.0*
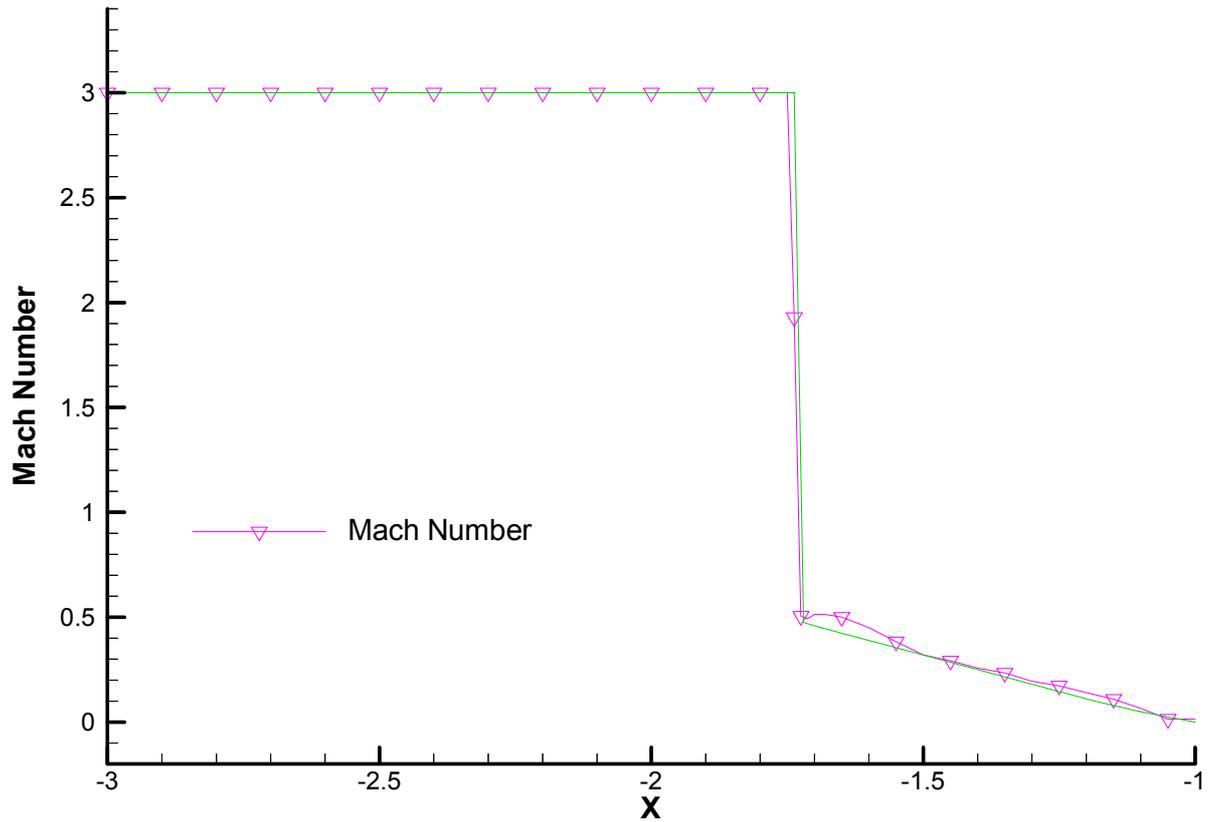
Figure 10.3.8: *Mach Number Profile for Adapted Quadrant, y = 0.0*
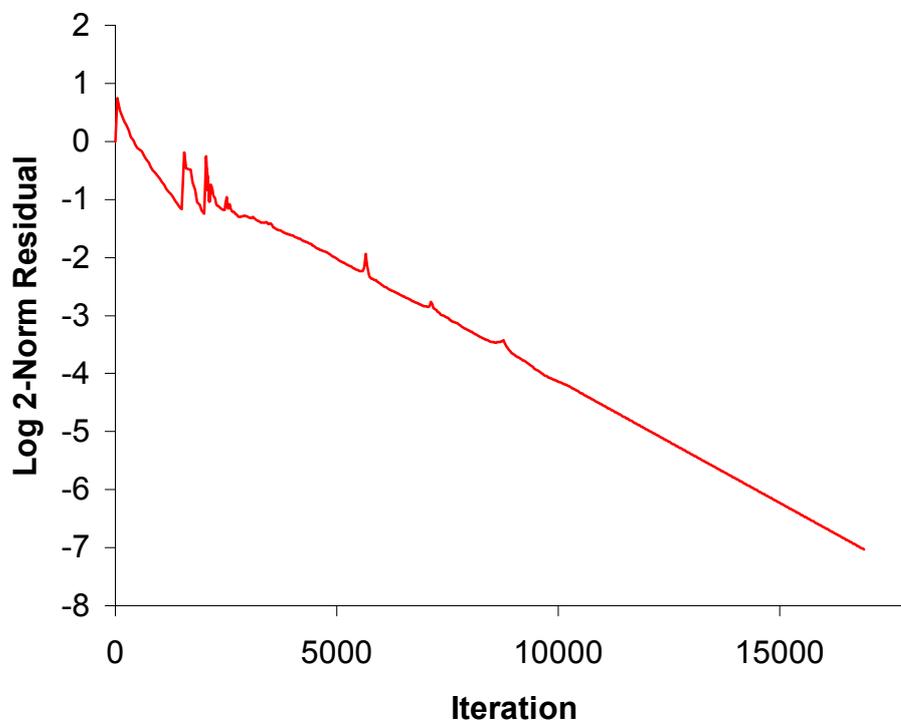
Figure 10.3.9: *Log 2-Norm Residual History for Quadrant*

10.4   Subsonic Nozzle

The purpose of the subsonic nozzle is to test the symmetric properties of the Euler Equations when there are no shocks present. The nozzle is of the converging-diverging type, and at any x position along the nozzle the flow properties can be derived from the continuity equation and the isentropic relations. The flow in such a case will be symmetric about $y = 0.0$ due to the symmetric nature of the geometry and symmetric about $x = 2.0$ implying that flow from either the left or the right will give an identical solution.
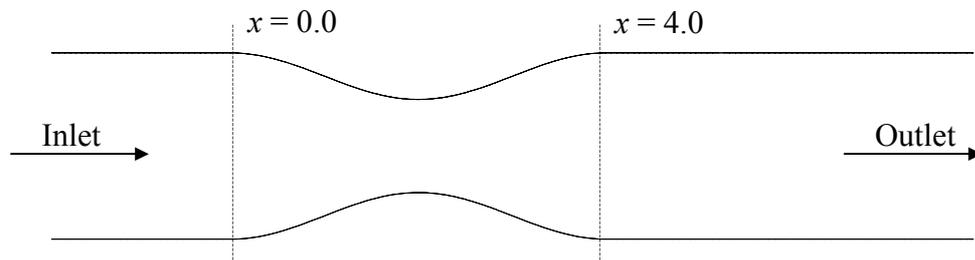
Figure 10.4.1: *Converging-Diverging Nozzle*

The section defined by $0.0 \leq x \leq 4.0$ is constructed from a cosine wave between the limits [0, 2π].

The inlet conditions for the current case are:

$$\begin{bmatrix} M \\ \rho \\ p \end{bmatrix}_1 = \begin{bmatrix} 0.2 \\ 1.0 \\ 1.0 \end{bmatrix} \tag{10.4.1}$$

The inlet mass flow can be calculated, and from the continuity equation the mass flow rate at any cross-sectional area A_2 can be calculated, based on a 1-dimensional analysis:

$$\dot{m} = \rho_1 u_1 A_1 = \rho_2 u_2 A_2 \tag{10.4.2}$$

By making suitable rearrangements:

$$\dot{m} = \rho u A = \frac{p}{RT} u A \tag{10.4.3}$$

$$\dot{m} = p \frac{u}{\sqrt{\gamma RT}} \sqrt{\frac{\gamma}{RT_0}} \sqrt{\frac{T_0}{T}} A \qquad \qquad \textbf{(10.4.4)}$$

From the definition for the stagnation temperature ratio and Mach number:

$$\frac{T_0}{T} = \left(1 + \frac{(\gamma - 1)M^2}{2}\right) \qquad \text{and} \qquad M = \frac{u}{\sqrt{\gamma RT}} \qquad \textbf{(10.4.5)}$$

Substituting into the above gives:

$$\dot{m} = pM \sqrt{\frac{\gamma}{RT_0}} \left(1 + \frac{(\gamma - 1)M^2}{2}\right)^{\frac{1}{2}} A \qquad \textbf{(10.4.6)}$$

From the isentropic relations, the total pressure and density can be determined for the current case:

$$p_0 = p \left(1 + \frac{(\gamma - 1)M^2}{2}\right)^{\left(\frac{\gamma}{\gamma - 1}\right)} \qquad \textbf{(10.4.7)}$$

$$\rho_0 = \rho \left(1 + \frac{(\gamma - 1)M^2}{2}\right)^{\left(\frac{1}{\gamma - 1}\right)} \qquad \textbf{(10.4.8)}$$

Substituting the above into the (10.4.6) gives mass flow rate in terms of total pressure:

$$\dot{m} = p_0 M \sqrt{\frac{\gamma}{RT_0}} \left(1 + \frac{(\gamma - 1)M^2}{2}\right)^{-3} A \qquad \textbf{(10.4.9)}$$

From the defined upstream conditions, p_0, T_0 and the mass flow rate can be calculated:

$$p_0 = 1.02828 \qquad T_0 = 0.003511 \qquad \dot{m} = 0.4662 \qquad \textbf{(10.4.10)}$$

Equation (10.4.9) can be easily solved for M for a given A using any mathematical package such as XMaple to give the following exact results:

x	*A*	*M*	ρ	*P*
-2	1.97	0.2	1.0	1.0
0	1.97	0.2	1.0	1.0
0.4122	1.87	0.21128	0.997703	0.996785
0.5868	1.77	0.22395	0.994981	0.992981
0.7363	1.67	0.23829	0.991724	0.988432
0.8694	1.57	0.25469	0.987771	0.982921
1.0	1.47	0.27363	0.982909	0.976155
1.1237	1.37	0.29580	0.976823	0.967703
1.2579	1.27	0.32217	0.969044	0.956932
1.4074	1.17	0.35417	0.958847	0.942864
1.5864	1.07	0.39408	0.945021	0.923886
2.0	0.97	0.44578	0.925412	0.897159
2.4136	1.07	0.39408	0.945021	0.923886
2.5926	1.17	0.35417	0.958847	0.942864
2.7421	1.27	0.32217	0.969044	0.956932
2.8763	1.37	0.29580	0.976823	0.967703
3.0	1.47	0.27363	0.982909	0.976155
3.1306	1.57	0.25469	0.987771	0.982921
3.2637	1.67	0.23829	0.991724	0.988432
3.4132	1.77	0.22395	0.994981	0.992981
3.5878	1.87	0.21128	0.997703	0.996785
4.0	1.97	0.2	1.0	1.0
8.0	1.97	0.2	1.0	1.0

Table 10.4.1: *Flow Properties for a Given Cross-Sectional Area*

It should be noted that these areas correspond to the co-ordinates between the two outer layers of points of the mesh and not the actual boundary of the mesh.

It can immediately be seen from the Mach number contours in Figure 10.4.4 that the unadapted mesh does not give a symmetrical solution; however, a single level of adaption, Figure 10.4.3, seems to solve this and can be seen in Figure 10.4.5. Figure 10.4.6, Figure 10.4.7 and Figure 10.4.8 show the pressure, density and Mach number profiles for the adapted solution compared to the exact values given above. The profiles compare very well. However, in all cases the exact solution sits slightly inside the computed profile. This is due to the two-dimensional effects of the computed solution. It can be seen from Figure 10.4.5 that the contours are curved and the length of this curve is longer than corresponding cross-sectional area at that point. At the inlet, the pressure and density are high and the Mach number is low, the author attributes this slight deviation to the fact that the inlet to the domain is too short and significant pressure waves are present upstream of the contraction. Although the centreline Mach number is low, it can be seen in Figure 10.4.9 that the mean Mach number

across the throat is 0.443 compared to the exact value from Table 10.4.1 of 0.44578. This is a numerical error of 0.6%.

The residual history is given in Figure 10.4.10. Adaption was performed at iteration 2000. Although convergence continues, it does take many iterations due to the subsonic nature of the domain.
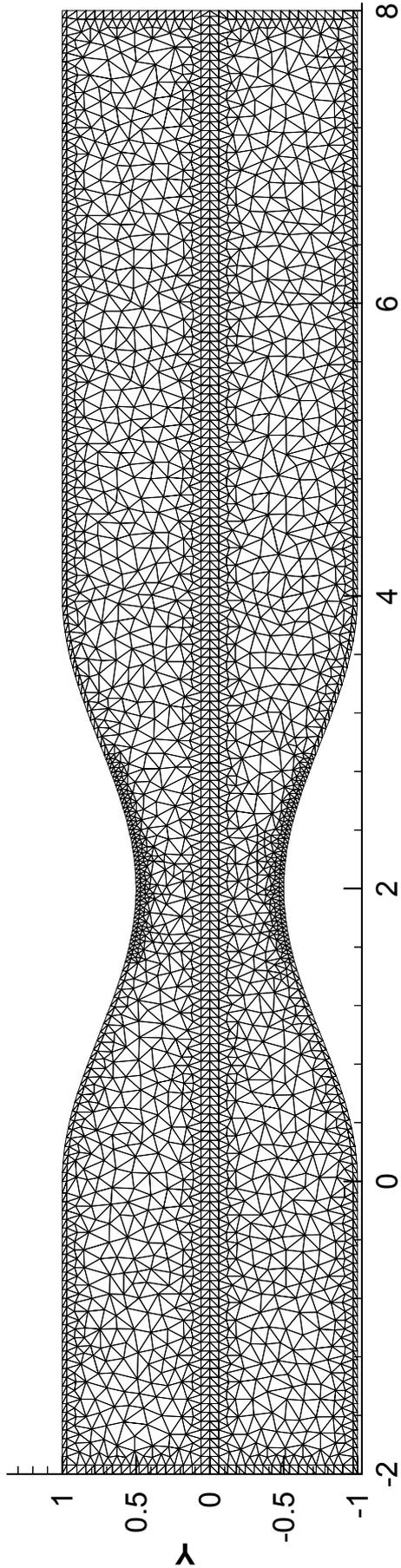
Figure 10.4.2: *Computational Mesh for Subsonic Nozzle*
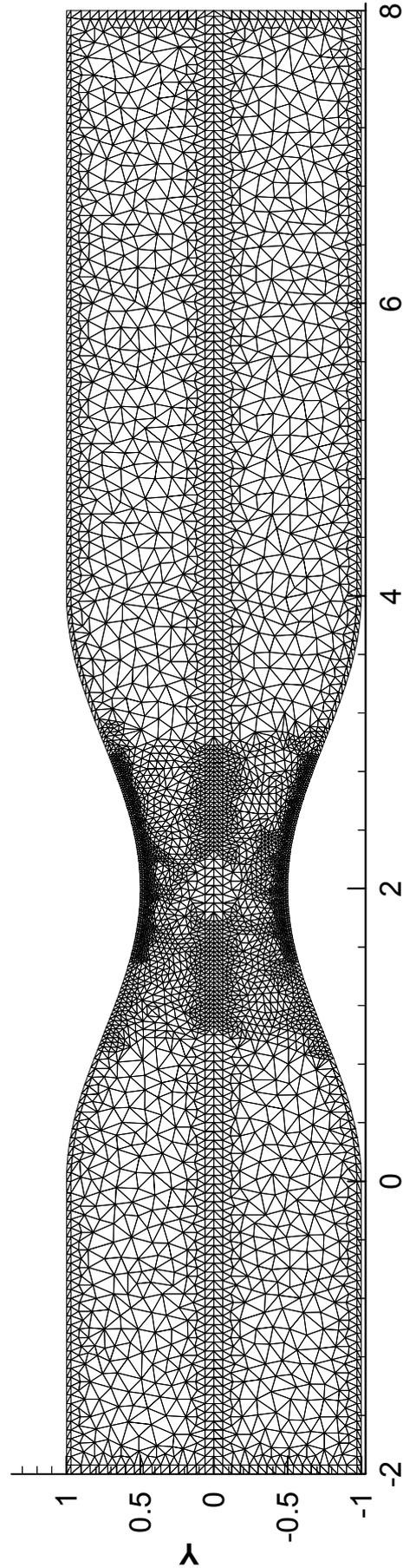
Figure 10.4.3: *Adapted Mesh for Subsonic Nozzle, 1 Adaptive Step*
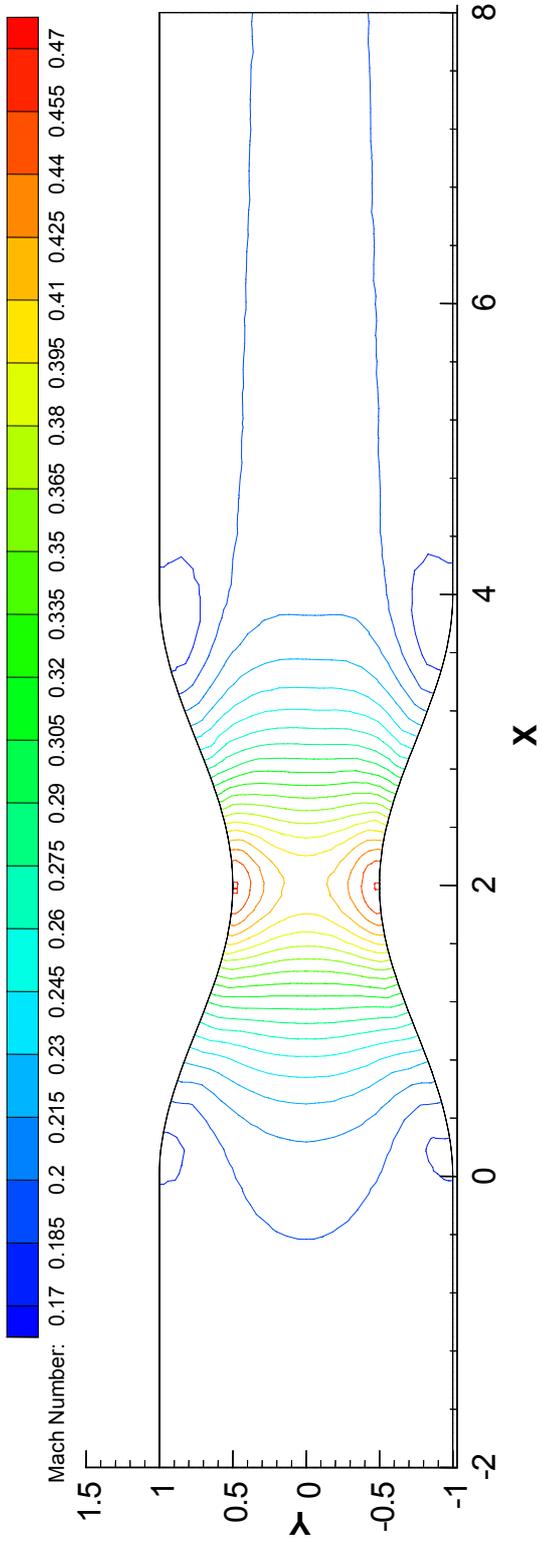
Figure 10.4.4: *Mach Number Contours for Subsonic Nozzle*
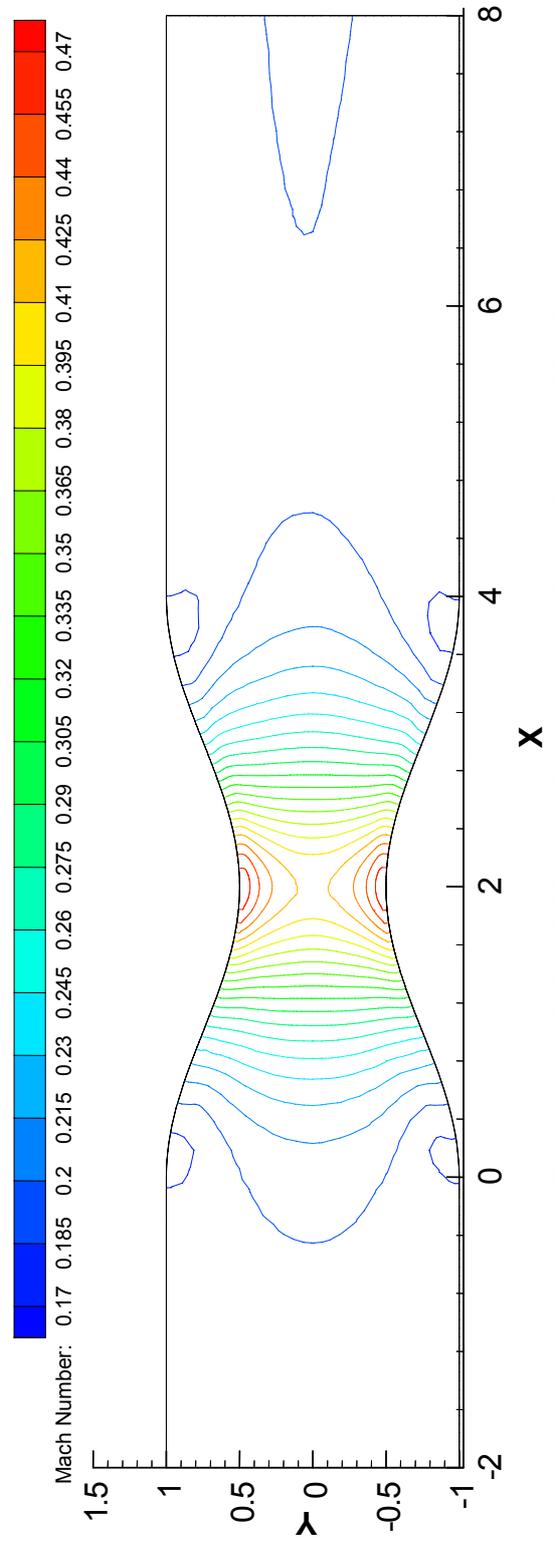
Figure 10.4.5: *Mach Number Contours for Adapted Subsonic Nozzle*
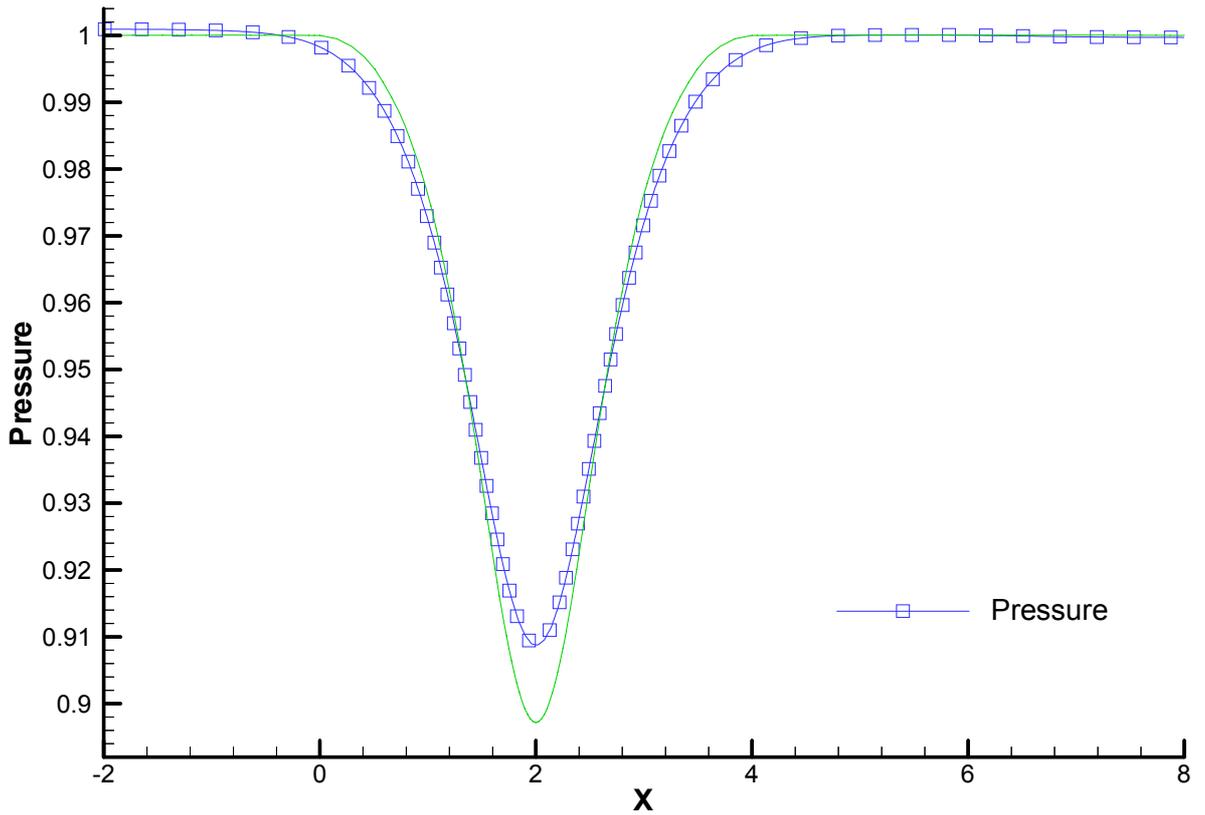
Figure 10.4.6: *Pressure Profile for Adapted Subsonic Nozzle, y = 0.0*
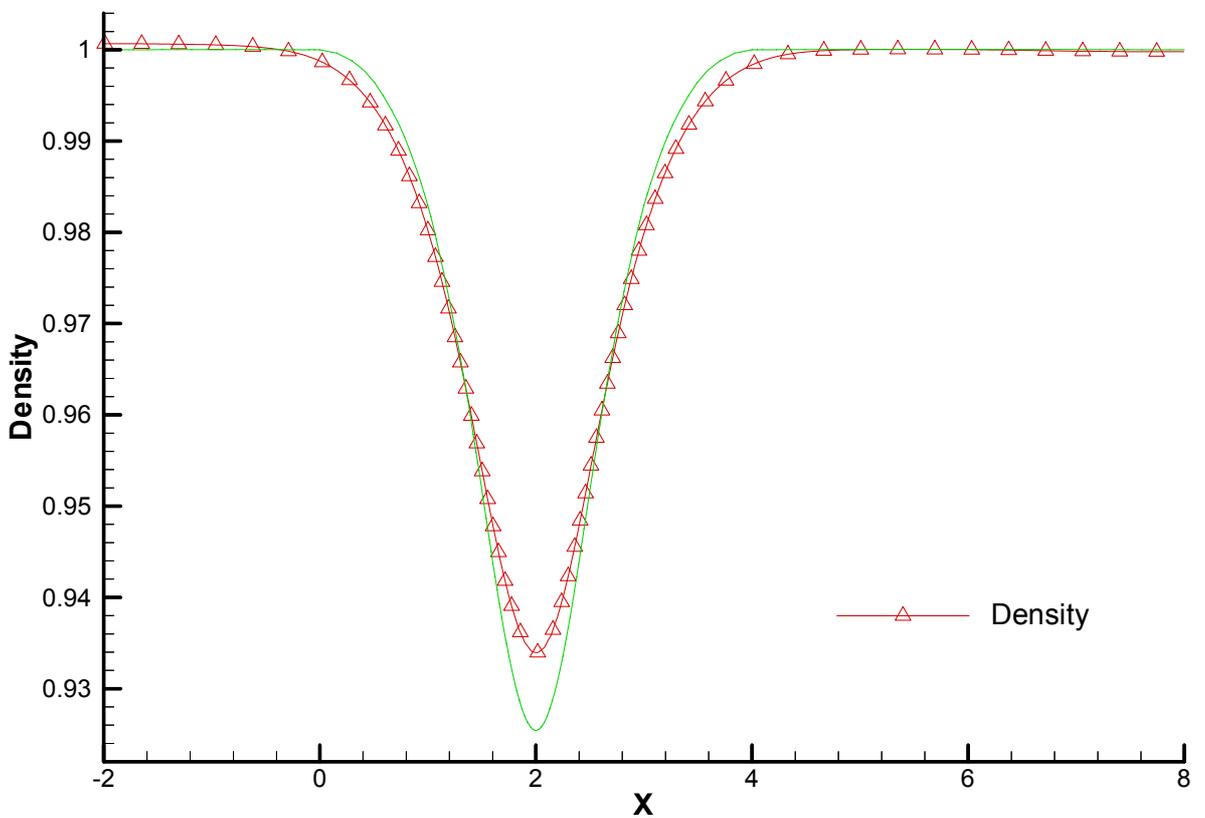
Figure 10.4.7: *Density Profile for Adapted Subsonic Nozzle, y = 0.0*
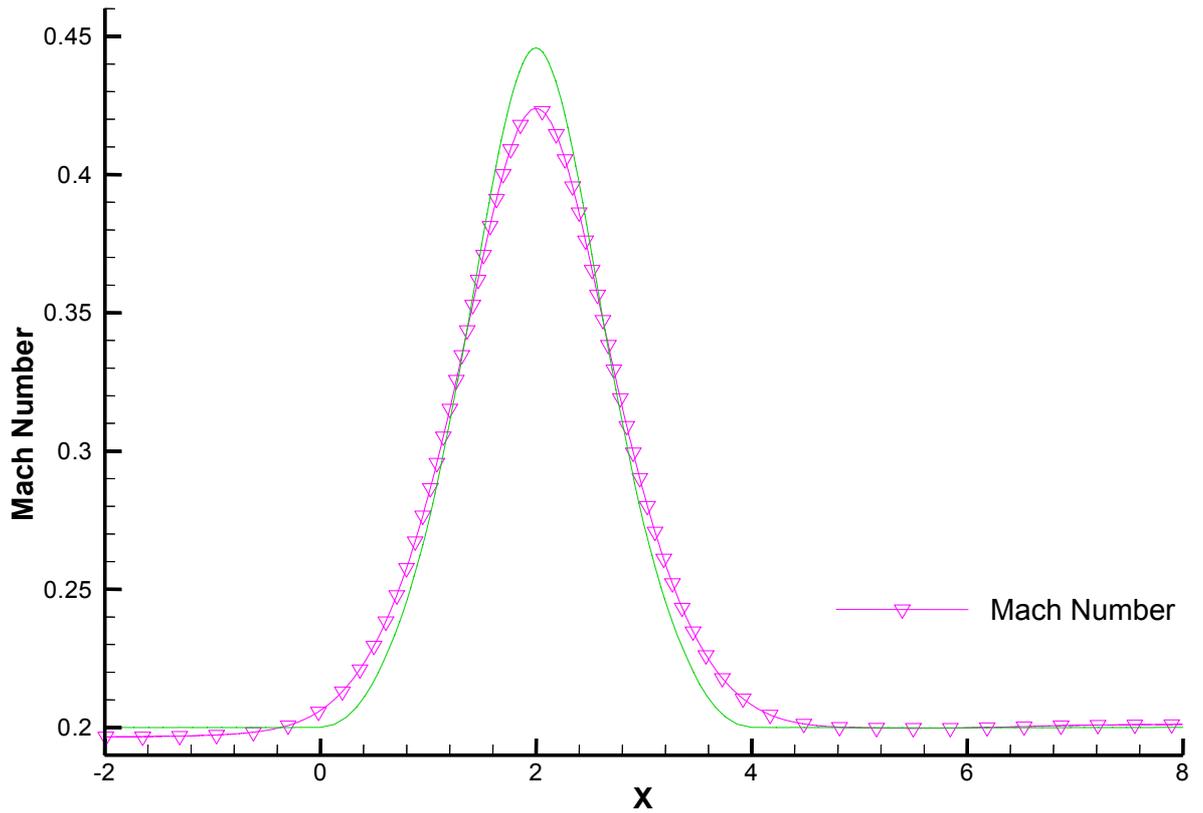
Figure 10.4.8: *Mach Number Profile for Adapted Subsonic Nozzle, y = 0.0*
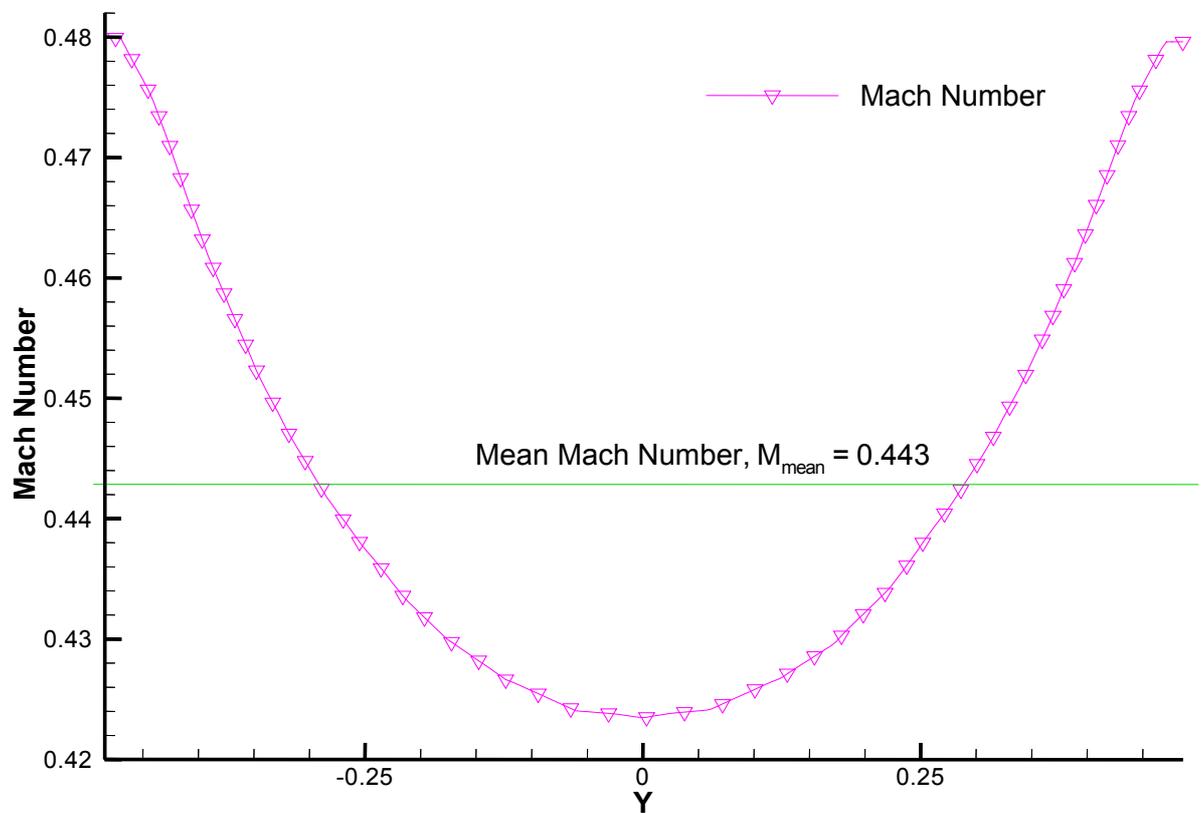
Figure 10.4.9: *Mach Number Profile for Adapted Subsonic Nozzle - Nozzle Throat, x = 2.0*
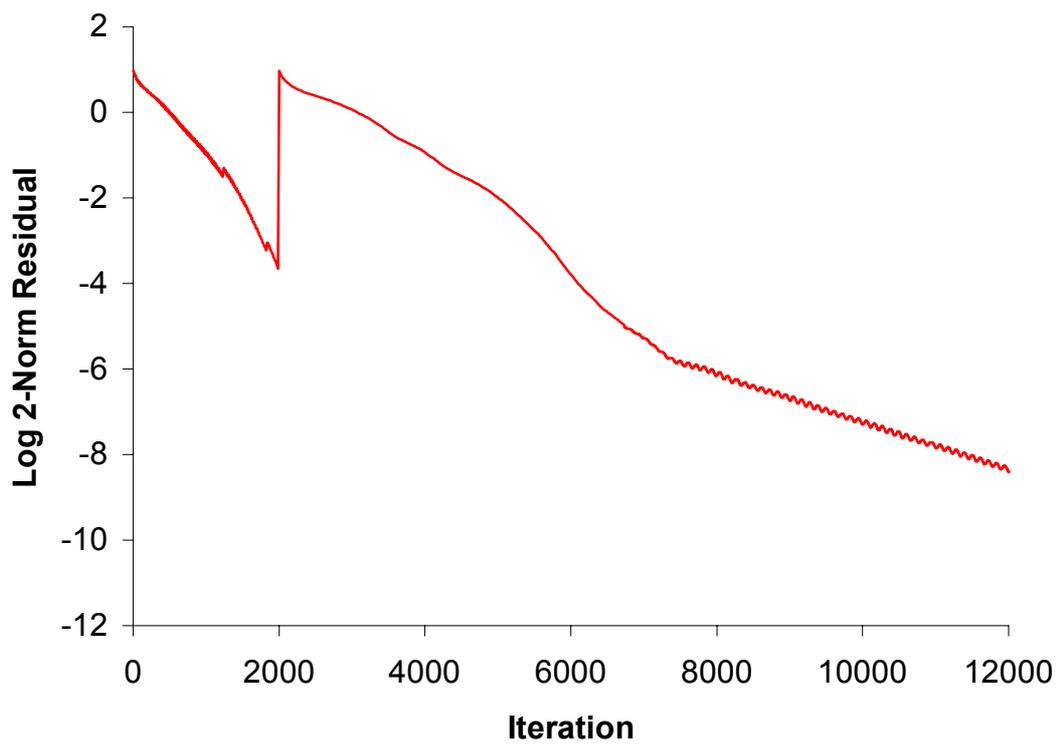
Figure 10.4.10: *Log 2-Norm Residual History for Subsonic Nozzle*

10.5   Transonic Nozzle

This is an amalgamation of the normal shock and the subsonic nozzle test cases. By increasing the inlet Mach number to the nozzle to the maximum allowable for the given area ratio, the downstream static pressure can be decreased to facilitate the formation of a shock in the diverging section of the nozzle.
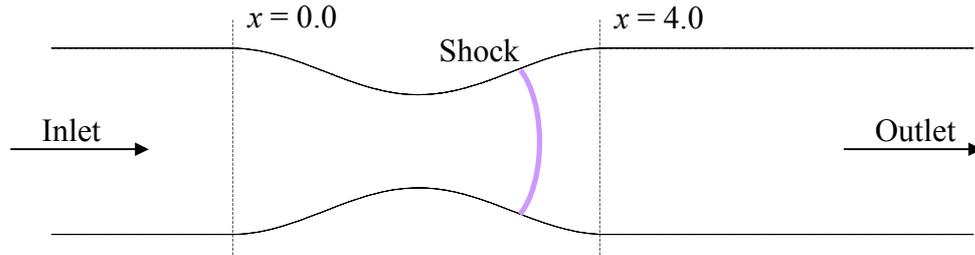
Figure 10.5.1: *Converging-Diverging Nozzle*

The inlet conditions for the test case are given as:

$$\begin{bmatrix} M \\ p \\ \rho \end{bmatrix}_1 = \begin{bmatrix} 0.3 \\ 1.0 \\ 1.0 \end{bmatrix} \tag{10.5.1}$$

Using the same principles as for the subsonic nozzle the flow properties from the inlet to the shock can be calculated from (10.5.2):

$$\dot{m} = 0.69928 = p_0 M \sqrt{\frac{\gamma}{RT_0}} \left(1 + \frac{(\gamma-1)M^2}{2} \right)^{-3} A \tag{10.5.2}$$

which can be solved for *M* using a relevant mathematical package. For the conditions downstream of the shock, equation (10.5.3) is employed (note the change in the power):

$$\dot{m} = 0.69928 = pM \sqrt{\frac{\gamma}{RT_0}} \left(1 + \frac{(\gamma-1)M^2}{2} \right)^{\frac{1}{2}} A \tag{10.5.3}$$

This is convenient, as downstream conditions are defined by the exit static pressure, which in this case is $p_{static} = 0.6666$. The following table is thus generated for the given cross-sectional areas:

x	A	M	ρ	p
-2	1.97	0.3	1.0	1.0
0	1.97	0.3	1.0	1.0
0.4122	1.87	0.31814	0.994514	0.992328
0.5868	1.77	0.3388	0.987929	0.983142
0.7363	1.67	0.36262	0.979907	0.971984
0.8694	1.57	0.39046	0.96997	0.958211
1.0	1.47	0.42361	0.957384	0.940851
1.1237	1.37	0.46405	0.940995	0.91838
1.2579	1.27	0.51513	0.918808	0.888207
1.4074	1.17	0.58338	0.886898	0.845324
1.5864	1.07	0.68545	0.835329	0.777321
2.0	0.97	0.95126	0.689865	0.594661
2.4136	1.07	1.38218	0.465622	0.342962
2.5926	1.17	1.54603	0.393688	0.271153
2.7421	1.27	1.67220	0.344413	0.22486
2.8763	1.37	1.77782	0.307244	0.191636
3.0	1.47	1.86983	0.277795	0.166423
3.1306	1.57	1.95192	0.253721	0.146588
3.2637	1.67	2.02635	0.233595	0.130572
3.4132	1.77	0.515	0.659417	0.637472
3.5878	1.87	0.47709	0.671341	0.653669
4.0	1.97	0.4453	0.680849	0.666665
8.0	1.97	0.4453	0.680849	0.666666

Table 10.5.1: *Flow Properties for a Given Cross-Sectional Area*

The above equations can be used irrespective of whether the flow is isentropic or not, as the total temperature will be constant in all cases. The non-adapted mesh is the same as for the subsonic nozzle. The adapted mesh can be seen in Figure 10.5.3. The adapted mesh gives a normal shock with very high definition, Figure 10.5.5, compared to the non-adapted case, Figure 10.5.4. The pressure, density and Mach number profiles along the centre-line, $y = 0.0$, are given in Figure 10.5.8, Figure 10.5.6, and Figure 10.5.7. The green line denotes the exact profiles based on a one-dimensional analysis. The computed profiles are in good agreement and show the correct trends in their shape. The downstream conditions show a loss in Mach number and density behind the shock. This is due to the down-stream static pressure being fixed. The entropy change therefore manifests itself as a change in density.
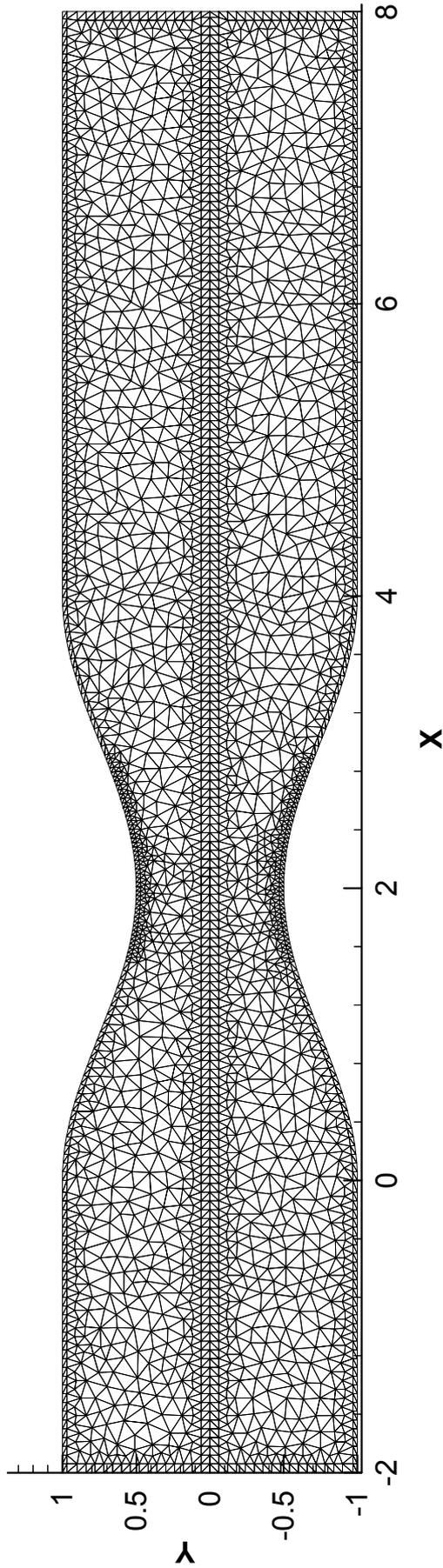
Figure 10.5.2: *Computational Mesh for Transonic Nozzle*
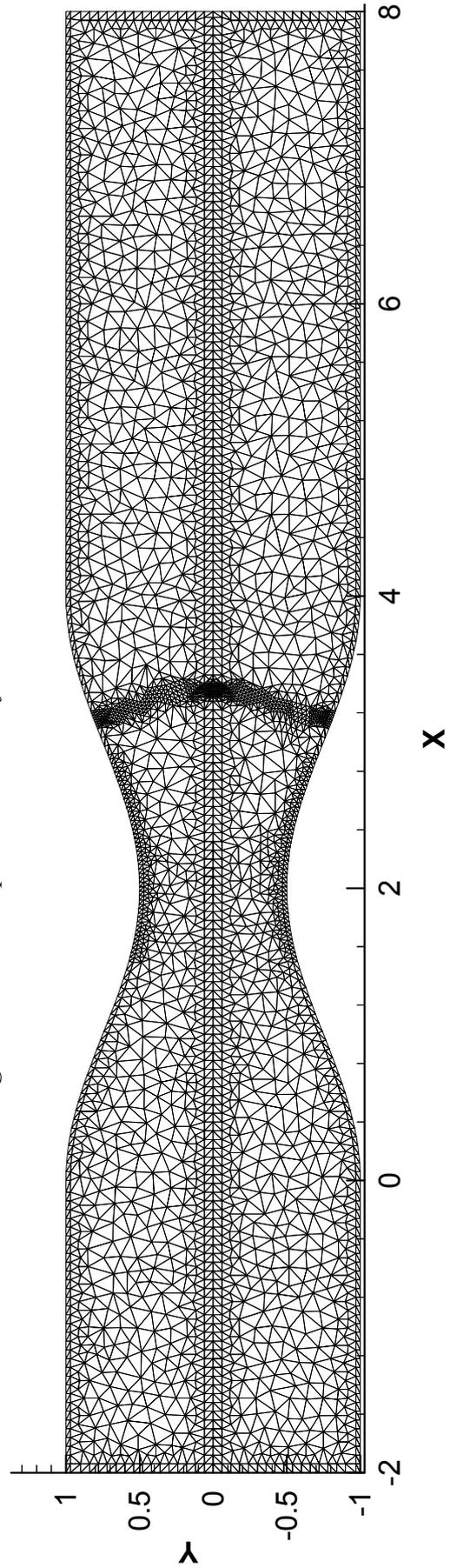
Figure 10.5.3: *Adapted Mesh for Transonic Nozzle, 2 Adaptive Steps*
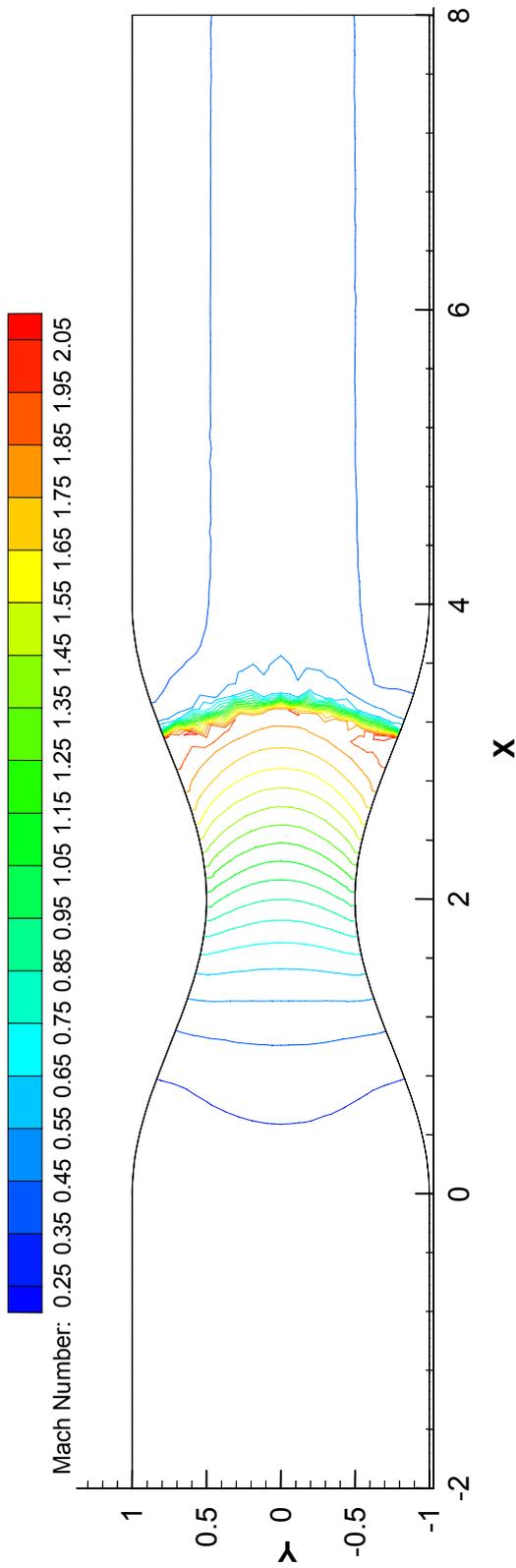
Figure 10.5.4: *Mach Number Contours for Transonic Nozzle*
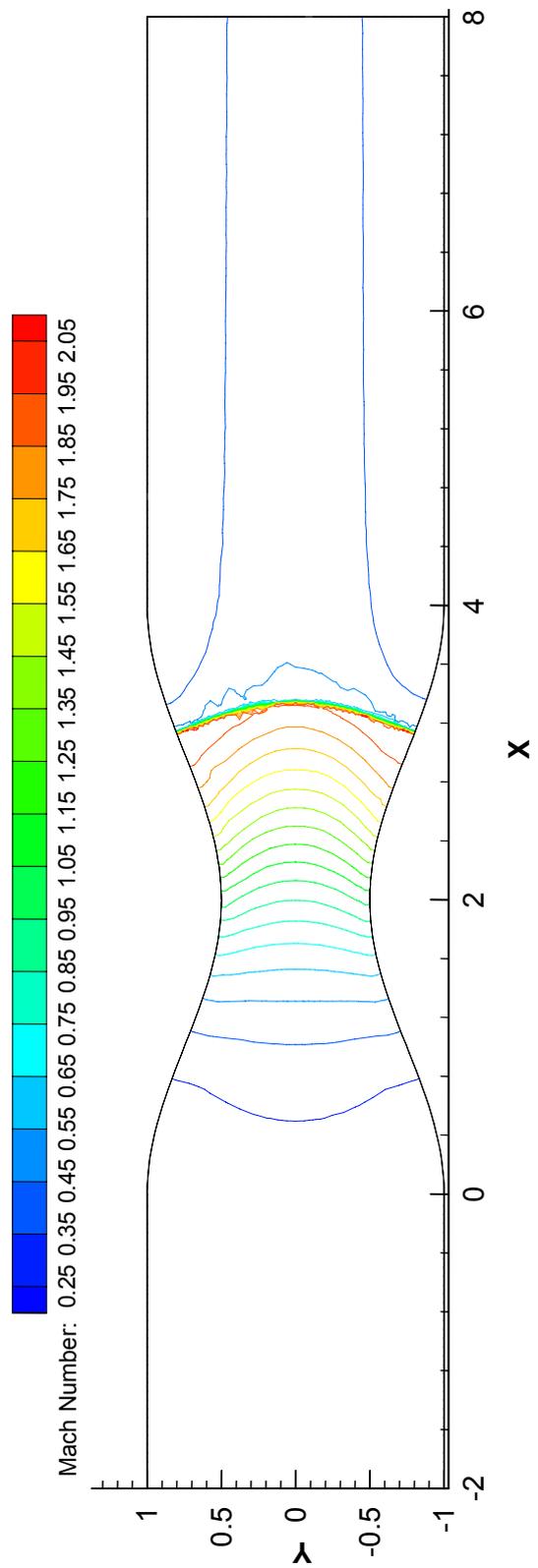
Figure 10.5.5: *Mach Number Contours for Adapted Transonic Nozzle*
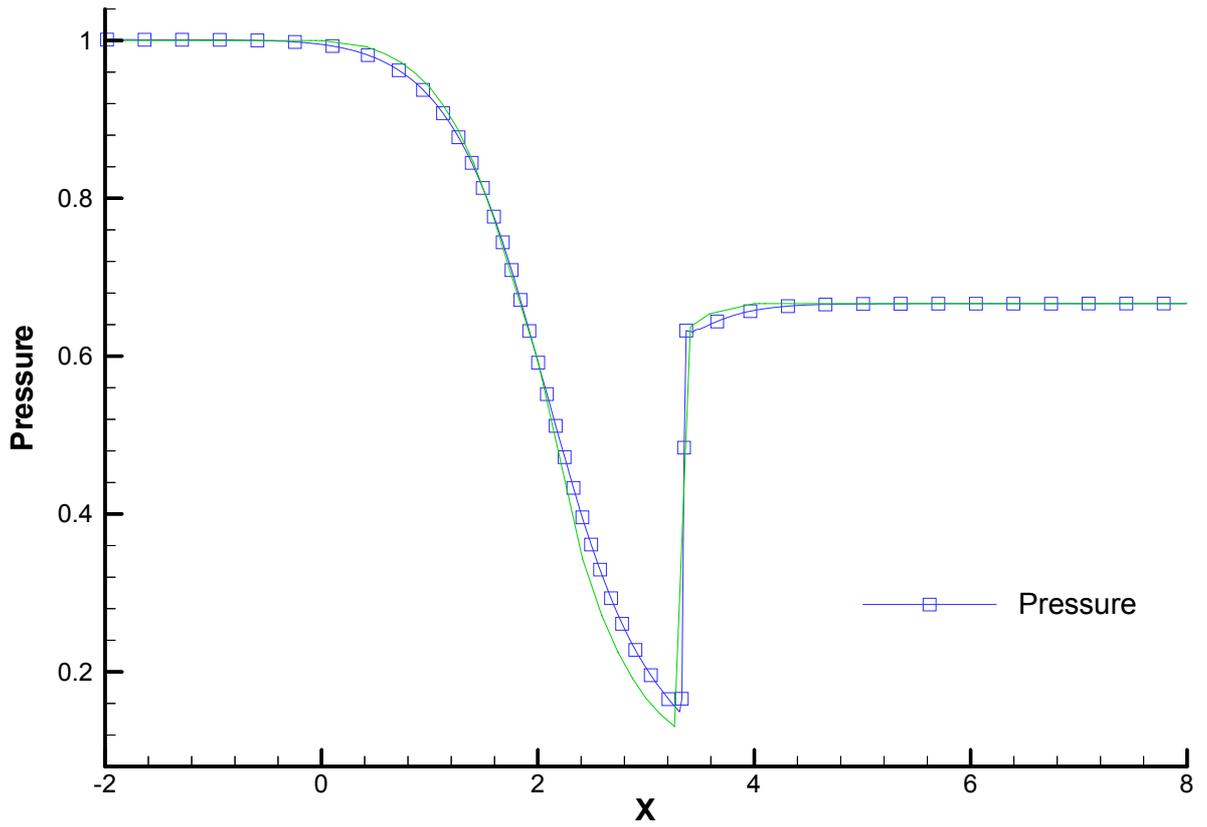
Figure 10.5.6: *Pressure Profile for Adapted Transonic Nozzle, y = 0.0*
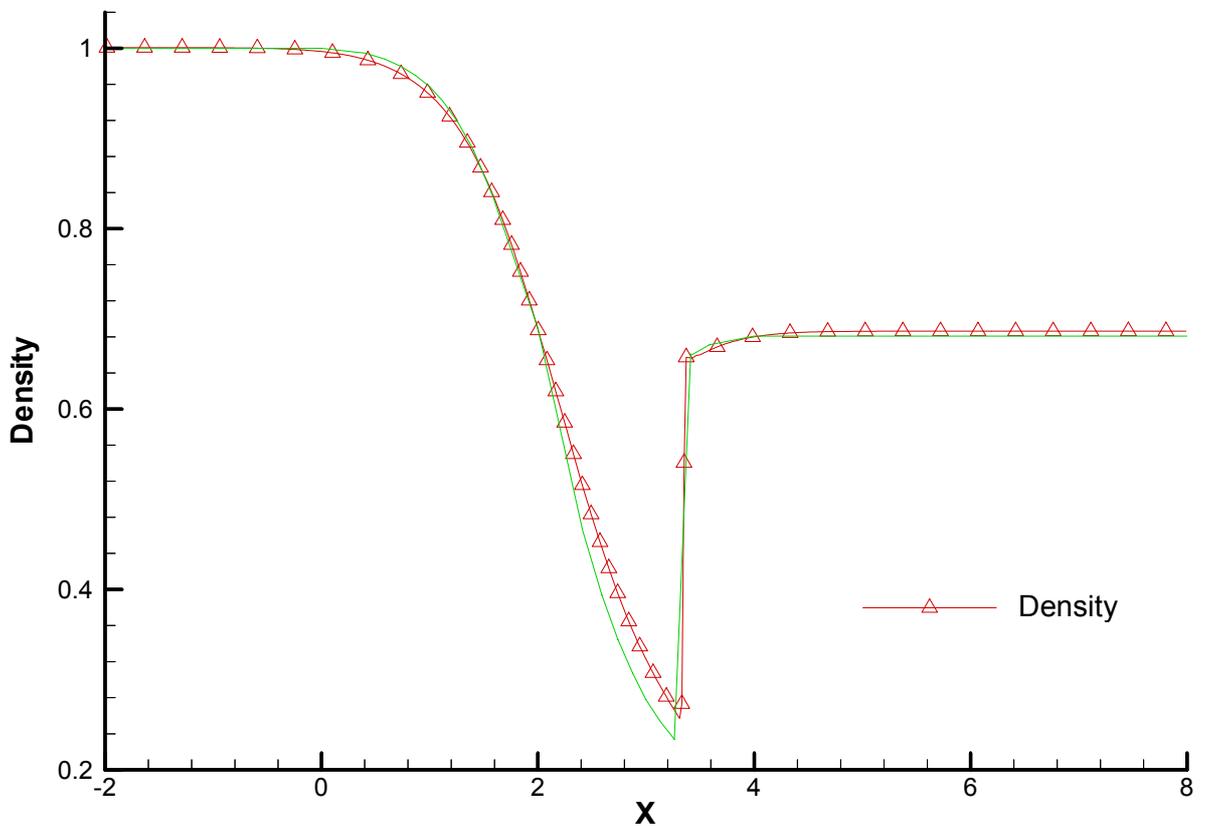
Figure 10.5.7: *Density Profile for Adapted Transonic Nozzle, y = 0.0*
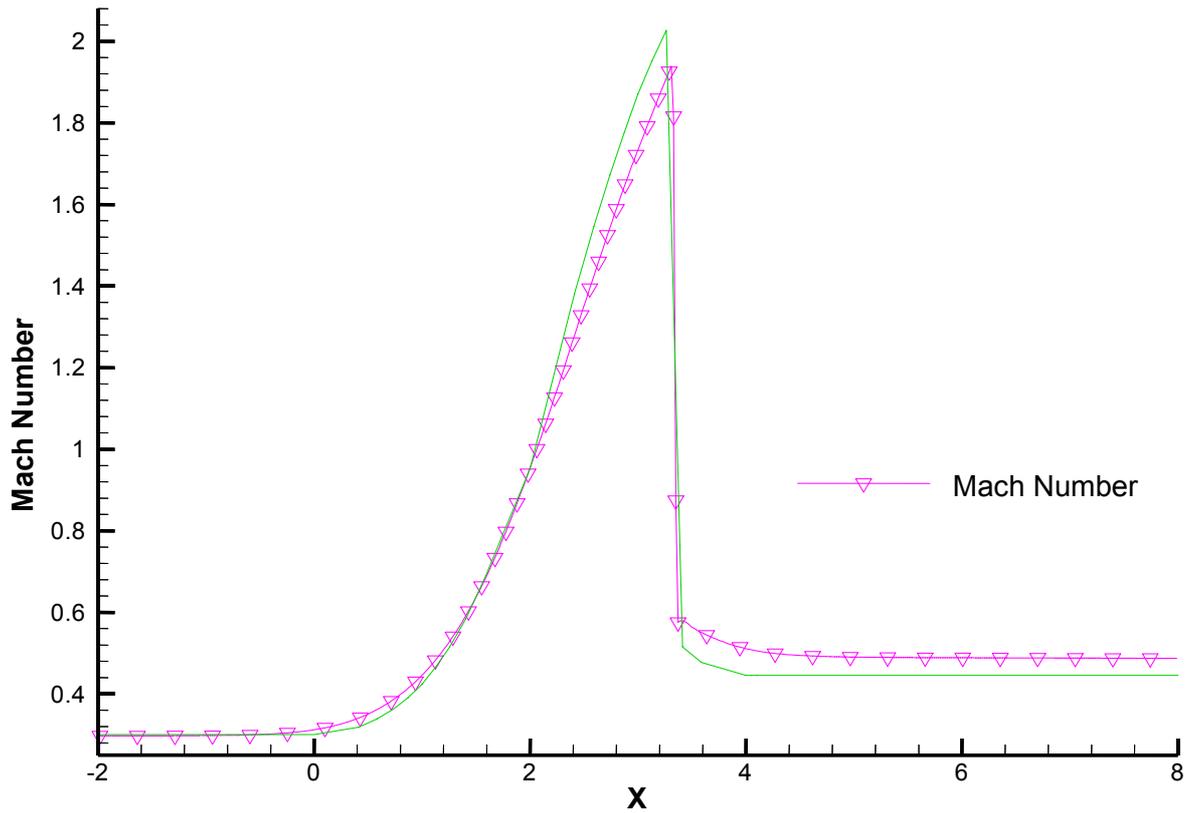
Figure 10.5.8: *Mach Number Profile for Adapted Transonic Nozzle, y = 0.0*
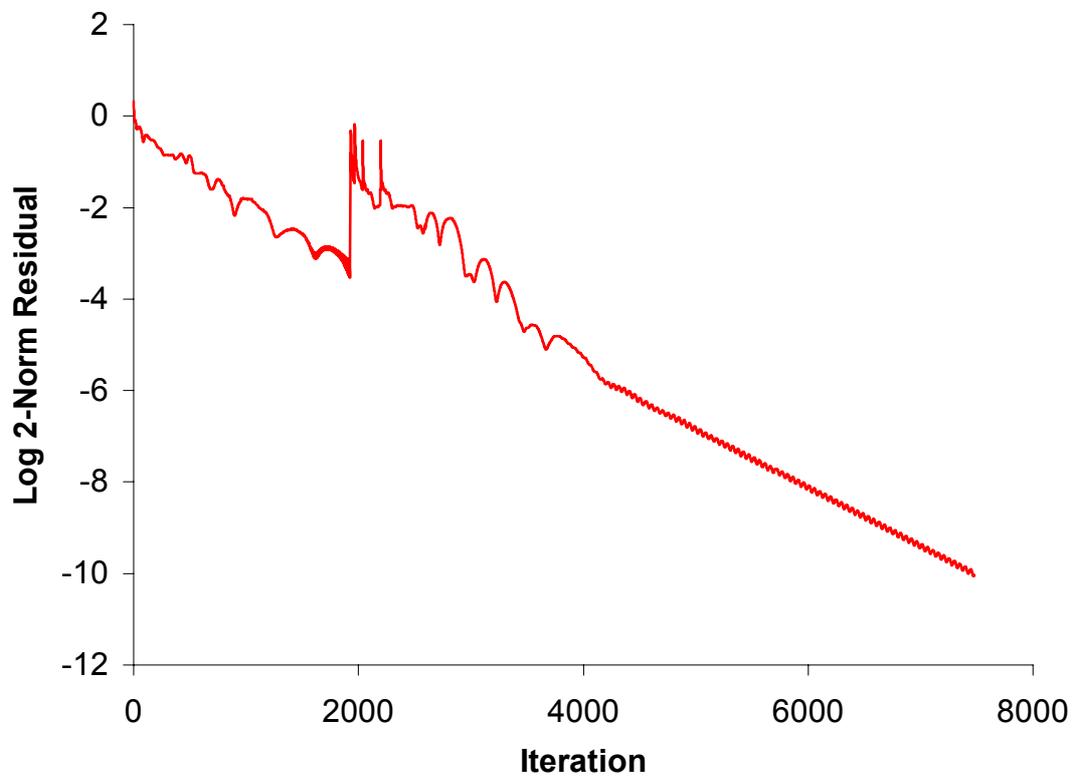
Figure 10.5.9: *Log 2-Norm Residual History for Transonic Nozzle*

10.6   Unsteady Shock Tube

An unsteady shock tube is based on the principal of a Riemann Problem, Hirsch (1990). A one-dimensional tube is separated into two halves by a diaphragm; the left and right states contain fluid at different pressures and densities. On removing the diaphragm (at time t=0), the interaction between the two states will be defined by a rightward moving pressure wave, a contact discontinuity and a leftward moving expansion wave, travelling at velocities $U+a$, U and $U-a$ respectively.
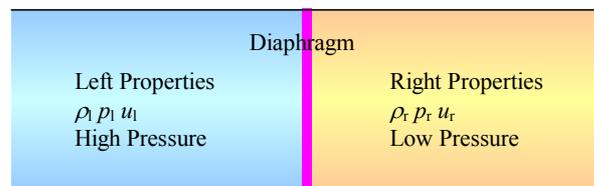
Figure 10.6.1: *Unsteady Shock Tube*

At any given time t>0 the solution can be separated into five discrete domains:
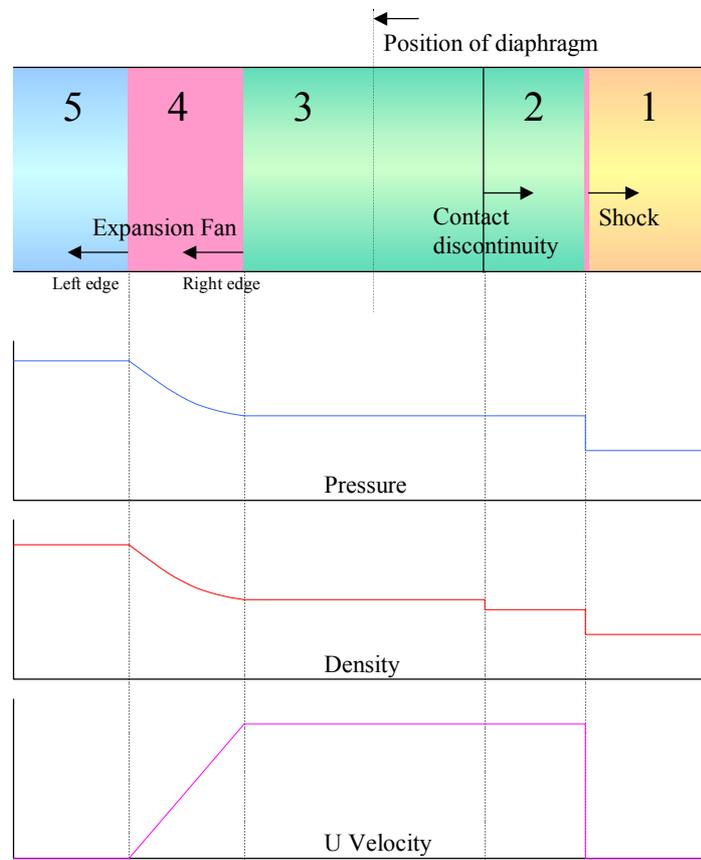
Figure 10.6.2: *Solution to the Shock Tube Problem*

For the current test case, the initial conditions are based on those of Hixon (1999):

$$\begin{bmatrix} p \\ \rho \\ u \end{bmatrix}_l = \begin{bmatrix} 4.4 \\ 3.66426 \\ 0.0 \end{bmatrix} \qquad \begin{bmatrix} p \\ \rho \\ u \end{bmatrix}_r = \begin{bmatrix} 1.0 \\ 1.2717 \\ 0.0 \end{bmatrix} \tag{10.6.1}$$

The density being calculated from $\rho = (\gamma p)^{1/\gamma}$

The mathematical solutions in each of the domains are:

Domain 1: As there is no influence from the approaching shock front, the fluid is still in the initial state, therefore:

$$\begin{bmatrix} p \\ \rho \\ u \end{bmatrix}_1 = \begin{bmatrix} 1.0 \\ 1.2717 \\ 0.0 \end{bmatrix} \tag{10.6.2}$$

The shock is moving to the right at $u = 1.48612$ ms^{-1}

Domain 2: Immediately behind the shock,

$$\begin{bmatrix} p \\ \rho \\ u \end{bmatrix}_2 = \begin{bmatrix} 2.17395 \\ 2.18488 \\ 0.621163 \end{bmatrix} \tag{10.6.3}$$

The contact discontinuity is moving at $u_{cd} = 0.621163$ ms^{-1}

Domain 3: behind the contact discontinuity

$$\begin{bmatrix} p \\ \rho \\ u \end{bmatrix}_3 = \begin{bmatrix} 2.17395 \\ 2.21447 \\ 0.621163 \end{bmatrix} \tag{10.6.4}$$

Domain 4: The area defined by the expansion fan. The properties for a given x position at time t are defined by the following equations:

$$u_4 = \frac{2}{(\gamma + 1)}\left(\frac{x}{t} + \sqrt{\frac{\gamma p_5}{\rho_5}}\right)$$
(10.6.5)

$$p_4 = p_5 \left(\frac{u_4 - \dfrac{x}{t}}{\sqrt{\dfrac{\gamma p_5}{\rho_5}}}\right)^{\left(\frac{2\gamma}{\gamma - 1}\right)}$$
(10.6.6)

$$\rho_4 = \frac{\gamma p_4}{\left(u_4 - \dfrac{x}{t}\right)^2}$$
(10.6.7)

The velocities at the boundaries of the expansion fan are:

left edge u_{le}= -1.29657 ms^{-1}

right edge u_{re} = -0.55118 ms^{-1}

Domain 5: There is no influence from the approaching expansion fan, so the fluid is still in the initial state:

$$\begin{bmatrix} p \\ \rho \\ u \end{bmatrix}_5 = \begin{bmatrix} 4.4 \\ 3.66426 \\ 0.0 \end{bmatrix}$$
(10.6.8)

The Shock Tube is included to verify that the solver captures the correct propagation of the characteristic waves.

Numerical calculations were performed with the 1st and 2nd implicit solvers over an interval of 2 seconds with a time step of 0.02 seconds. Results are presented for the solution at time $t =$ 2.0 seconds after 100 iterations. The diaphragm is located at $x = 4.0$.

The mesh was defined with the upper and lower boundaries as periodic to remove any two-dimensional effects. The inlet is on the left and the outlet on the right; this can be seen in Figure 10.6.3.

The 2nd order solver can immediately be seen to give a sharper defined shock than the 1st order solver as demonstrated by the pressure contours in Figure 10.6.5 and Figure 10.6.4 respectively. Looking more closely at the profiles for the 1st order solution, Figure 10.6.6, the expansion fan does follow the exact line very well. The contact discontinuity is present,

although smeared over several points, and the forward shock is well defined with no overshoot. The exact solution is represented by a green line where applicable. With regards to the 2nd order case, Figure 10.6.7, the solution follows the exact profiles much more closely than the 1st order case – the expansion fan is very well defined, the contact discontinuity is sharper and the forward shock is captured more accurately. However, there is now an overshoot present behind all of the moving waves. The author attributes this to dispersion within the solver. The same features can be seen in the 1st order case but only to a minor degree. It is assumed that the extension to the 2nd order time accurate scheme does not accommodate the TVD properties of the 1st order scheme.

Figure 10.6.8 shows the residual history for the inner iterations performed in advancing a global time-step. The solver currently requires a large number of inner iterations, and the increase in accuracy does not currently outweigh the necessary increase in processing.
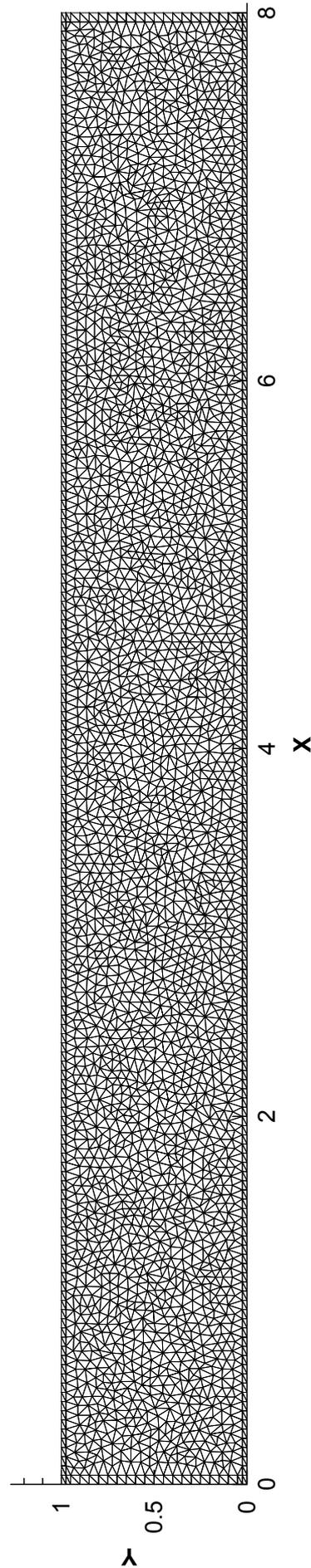
Figure 10.6.3: *Computational Mesh for Shock Tube*
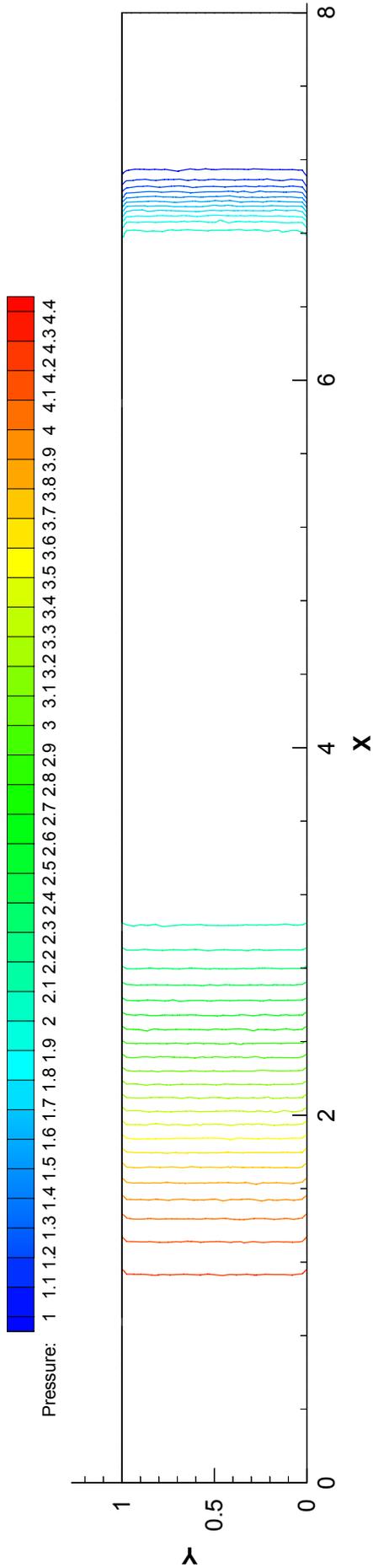
Figure 10.6.4: *Pressure Contours for 1st Order Shock Tube*
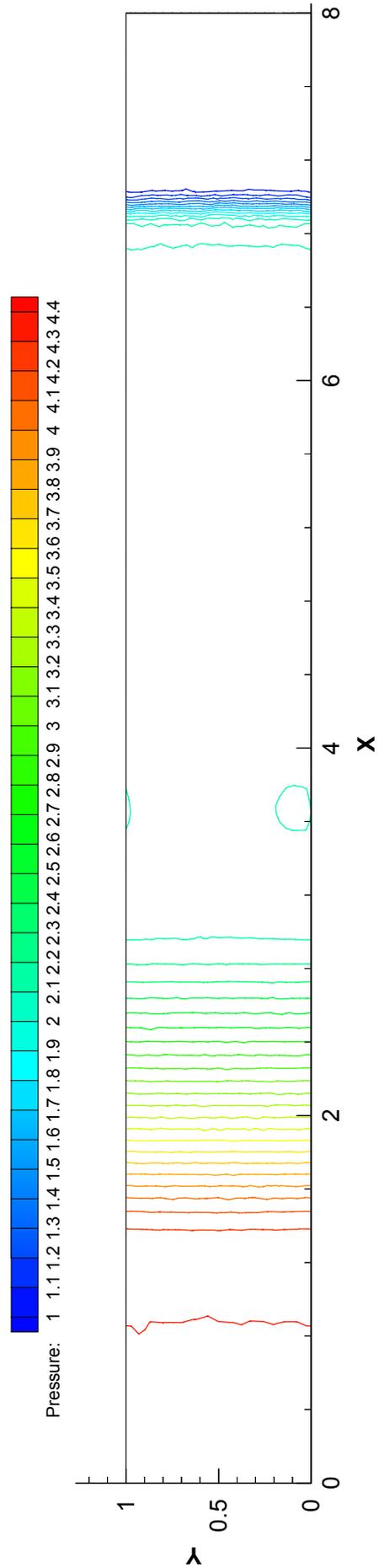
Figure 10.6.5: *Pressure Contours for 2nd Order Shock Tube*
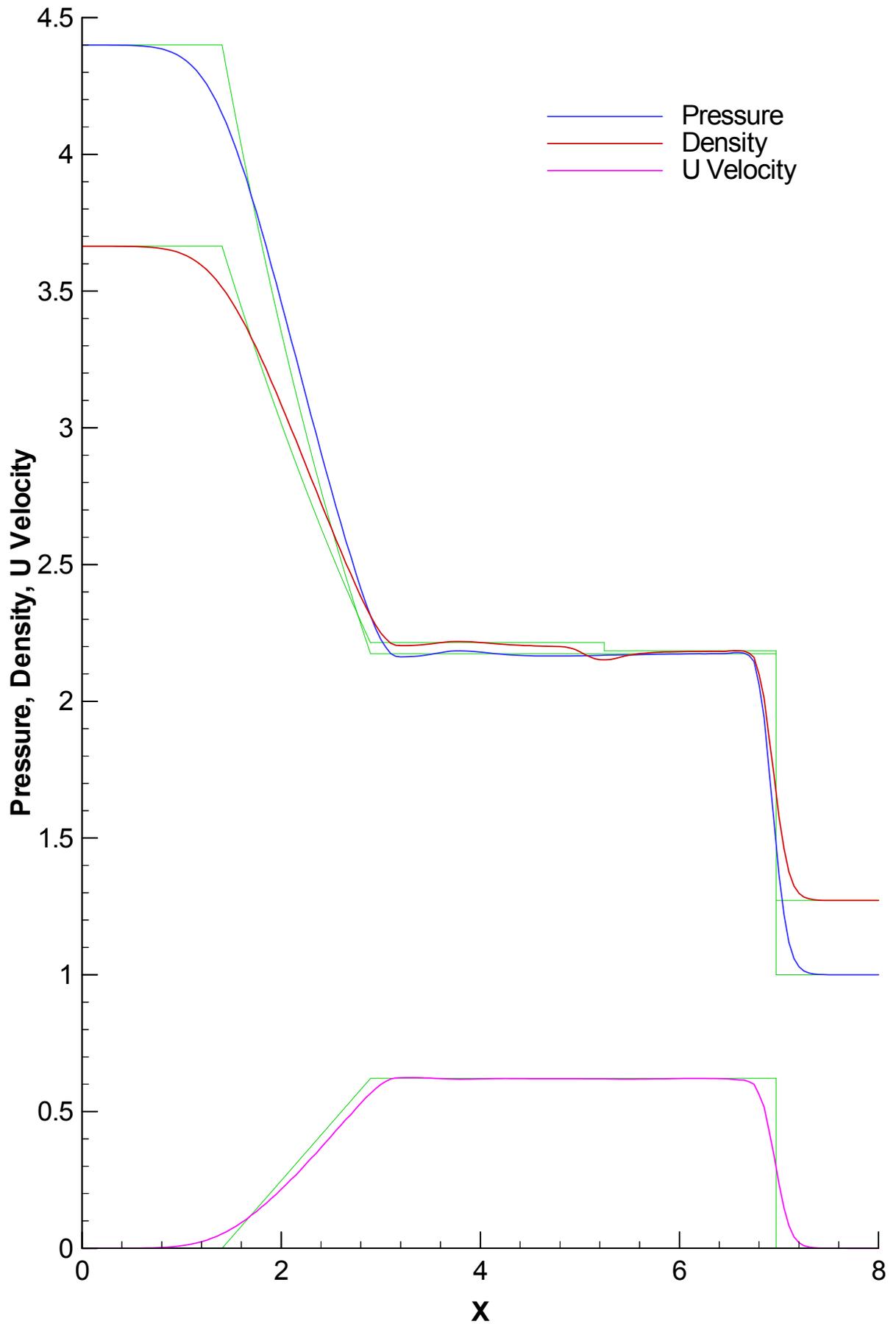
Figure 10.6.6: *Pressure, Density and U Velocity Profiles for 1ˢᵗ Order Shock Tube, y = 0.0*
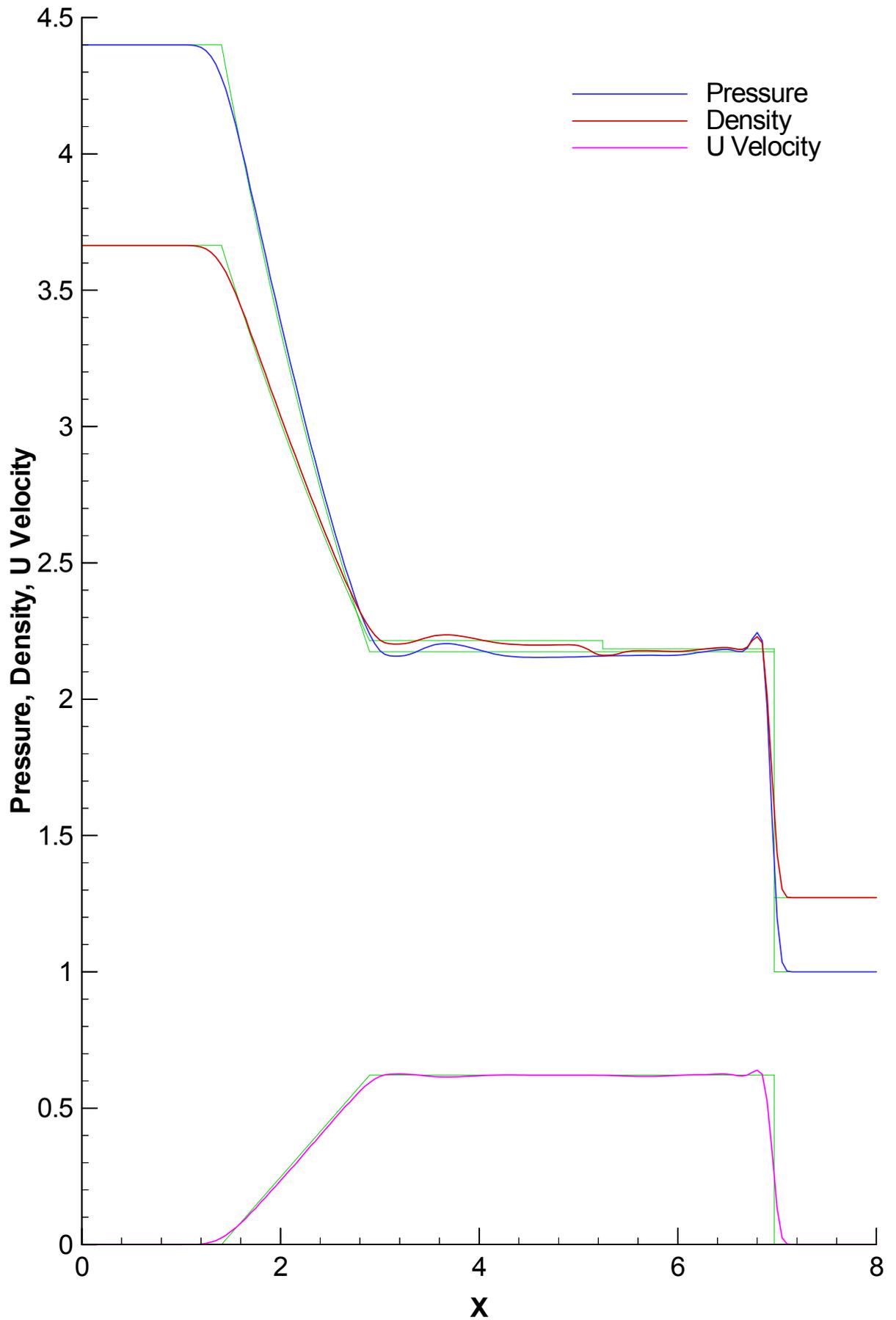
Figure 10.6.7: *Pressure, Density and U Velocity Profiles for 2nd Order Shock Tube, y = 0.0*
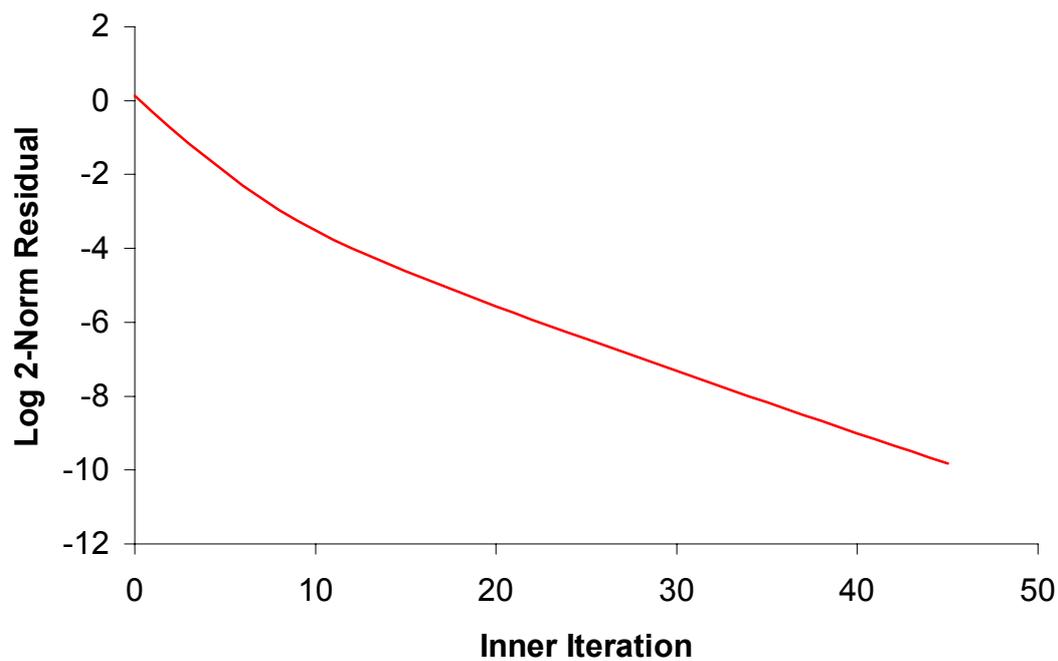
Figure 10.6.8: *Log 2-Norm Residual History for Inner Iterations of 2nd Order Shock Tube*

10.7   Unsteady Step

The last qualitative test case is the supersonic backward facing step. The purpose of the test is to evaluate the performance of the solver when computing the interaction between shear layers and shock waves. This is of particular importance as it simulates the flow in the trailing edge region of the turbine blade. The geometry for the test case can be seen in Figure 10.7.1.
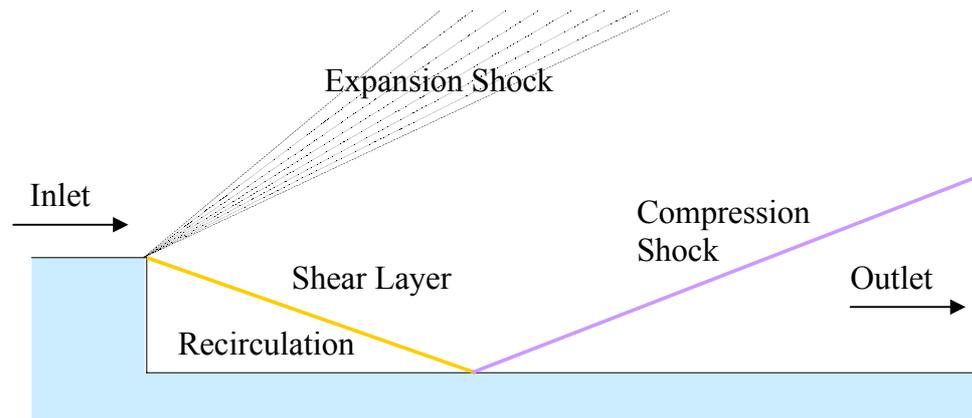
Figure 10.7.1: *Backward Facing Step*

The inlet conditions are:

$$\begin{bmatrix} M \\ p \\ \rho \end{bmatrix}_1 = \begin{bmatrix} 2.5 \\ 1.0 \\ 1.0 \end{bmatrix} \tag{10.7.1}$$

The computational mesh is given in Figure 10.7.2. The Mach number contours for the solution can be seen in Figure 10.7.3. The flow can be seen entering the left hand side of the domain. As the flow crosses the edge of the step, an expansion fan forms. The flow is turned towards the base of the step in a similar manner to the expansion corner, as described in chapter 10.2. The inlet Mach number defines the leading edge angle of the expansion fan. From characteristic tables, the leading edge angle $\alpha_1 = 23.58°$.

The trailing edge angle of the expansion fan is defined by the static pressure within the recirculation, Figure 10.7.4. This is not low enough that the flow is completely turned against the back of the step, hence, a shear layer develops. The flow below the shear layer is forced into a recirculation. The flow continues downstream, strikes the lower wall and is turned back on itself into the main flow. At this junction, a compression shock is formed. Analytical solutions are not available for this test case; however, the numerical results are compared with

the experimental data of Smith (1967). The numerical pressure profile along the lower wall is given in Figure 10.7.5. The results can be seen to compare favourably with the experimental results. The shapes of the two curves are very similar, although the numerical pressures are slightly underestimated. This is attributed to the numerical computation being inviscid. A viscous computation would reduce the wall velocity thereby increasing the wall static pressure, giving a more accurate correlation. The experimental values are given in Table 10.7.1.

X position	$\dfrac{p}{p_0}$
1.1875	0.01111
1.4584	0.01111
1.75	0.01481
2.037	0.02643
2.2963	0.0406
2.7962	0.0535
3.0377	0.05464
3.2453	0.05518
3.5472	0.05571
3.7925	0.05625
4.0392	0.05678
4.2353	0.0575
4.549	0.0575
4.7843	0.0575
5.0392	0.05786
5.2353	0.05786
5.549	0.05786
5.7843	0.05786
6.04	0.058528
7.04	0.058528

Table 10.7.1: *Experimental Pressure Readings along Lower Wall, Smith (1967)*
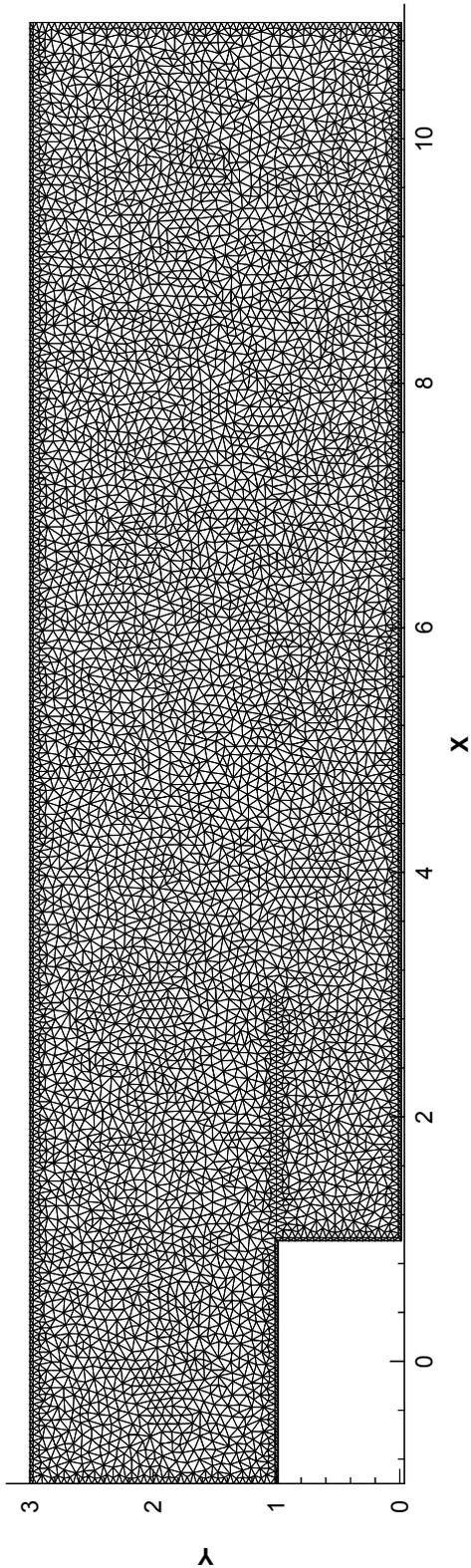
Figure 10.7.2: *Computational Mesh for Backward Facing Step*
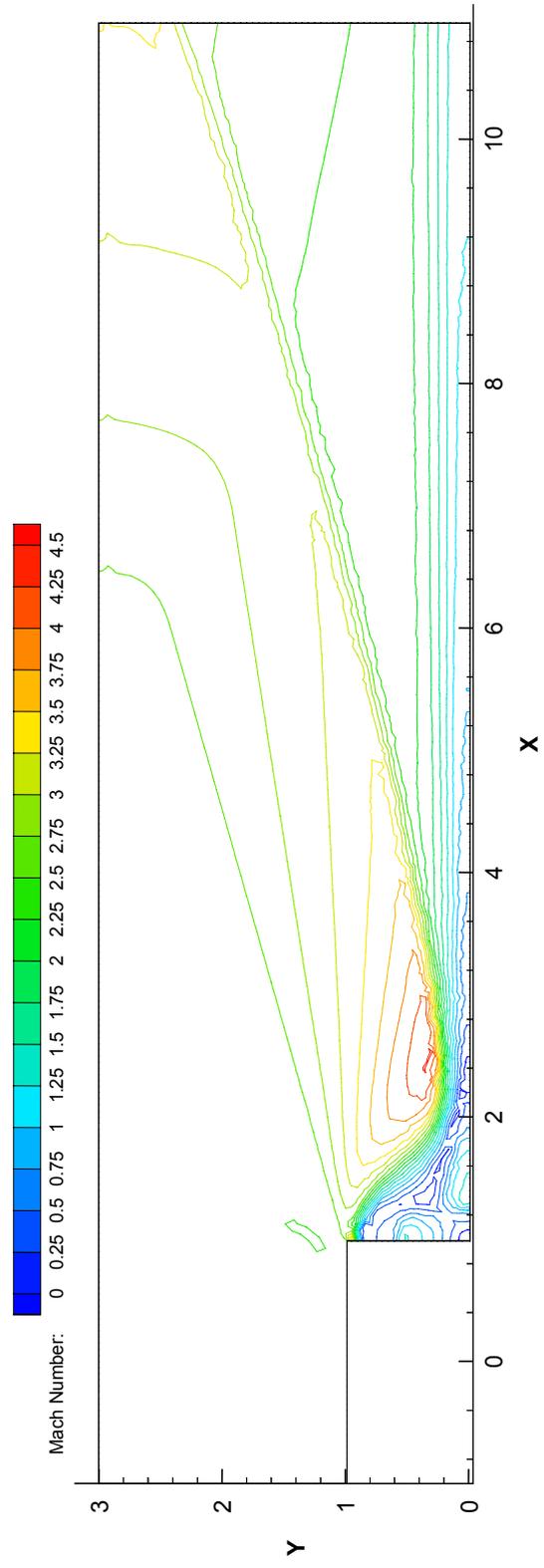
Figure 10.7.3: *Mach Number Contours for Backward Facing Step, ΔM = 0.25*
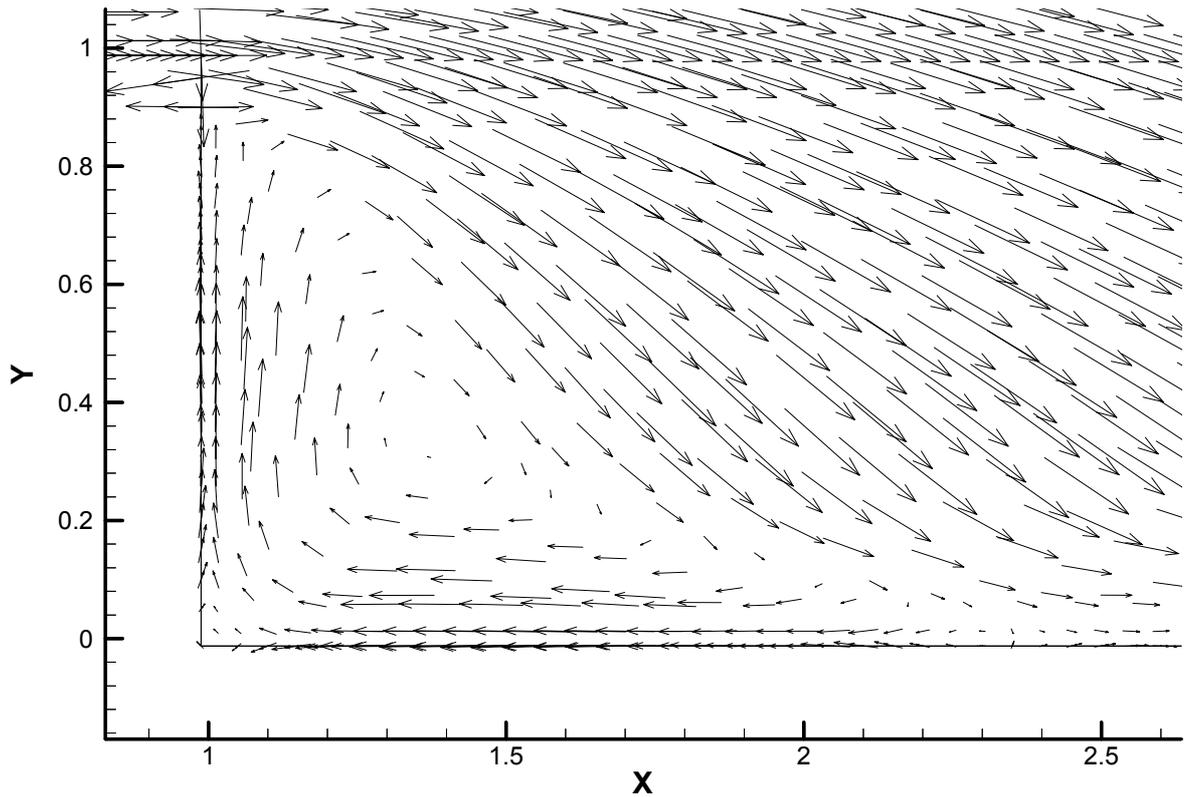
Figure 10.7.4: *Recirculation for Backward Facing Step*
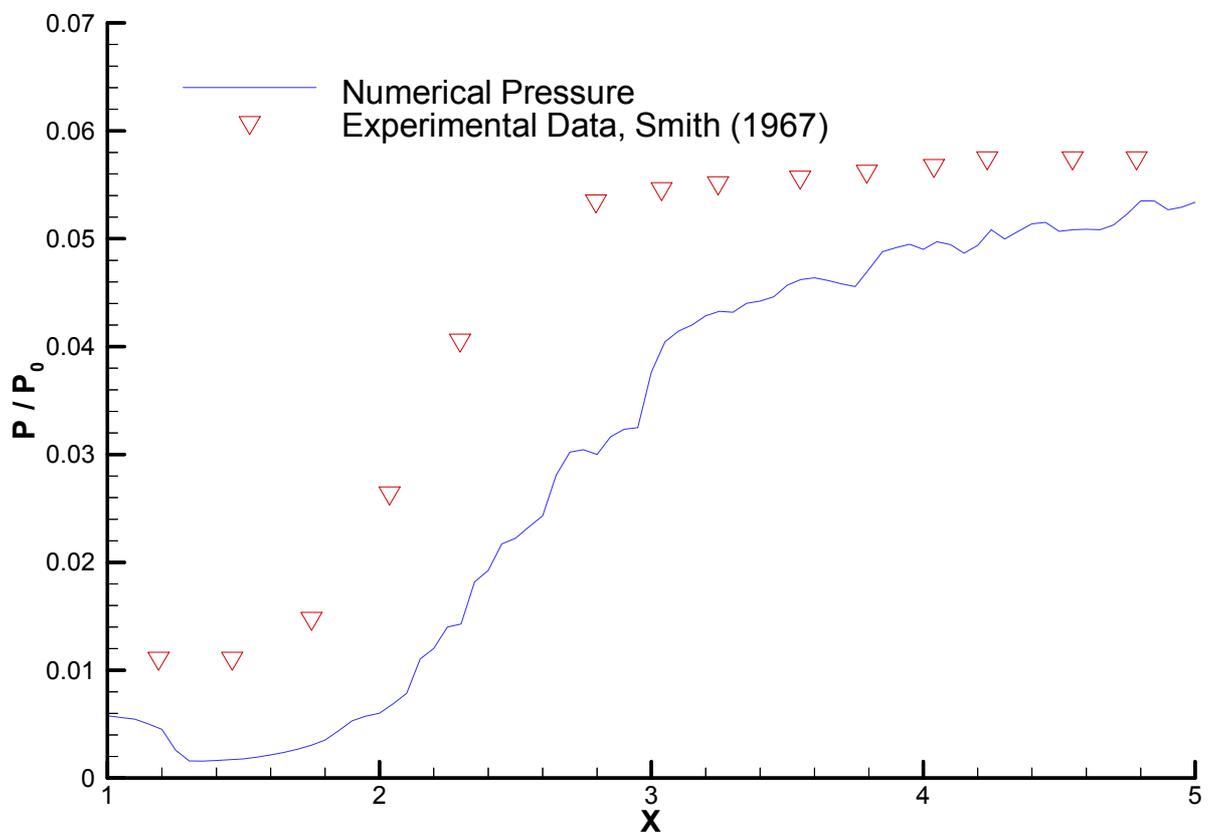
Figure 10.7.5: *Pressure Profile Along Lower Wall, y = 0.0*

10.8   Concluding Remarks

A series of test cases has been presented for the validation of the numerical solver. The range of tests covers the types of flow and conditions that may exist in transonic turbomachinery. It can be seen that the solver performs very well; it can be used confidently for the turbine blade and water table geometries on the following chapters.

11   TURBINE CASCADE GEOMETRY

The profile under investigation for the present study is the mean section of a first stage nozzle designed for a highly loaded turbine stage. The mean section is given below with details of the geometry dimensions. The complete co-ordinate set can be found in Appendix B.

Pressure ratio	2.3
Inlet Mach Number	0.112
Isentropic Exit Mach Number	1.16
Blade Chord	204mm
Axial Blade Chord	84.7mm
Stagger Angle	65.5°
Pitch	147.8mm
Span	112.8mm
Nozzle Turning Angle	76°
Nozzle Throat Opening	30.9mm
Suction surface Length	221.5mm
Pressure Surface Length	186.5mm
Trailing Edge Diameter	6.35mm
Inlet Metal Angle	-10°
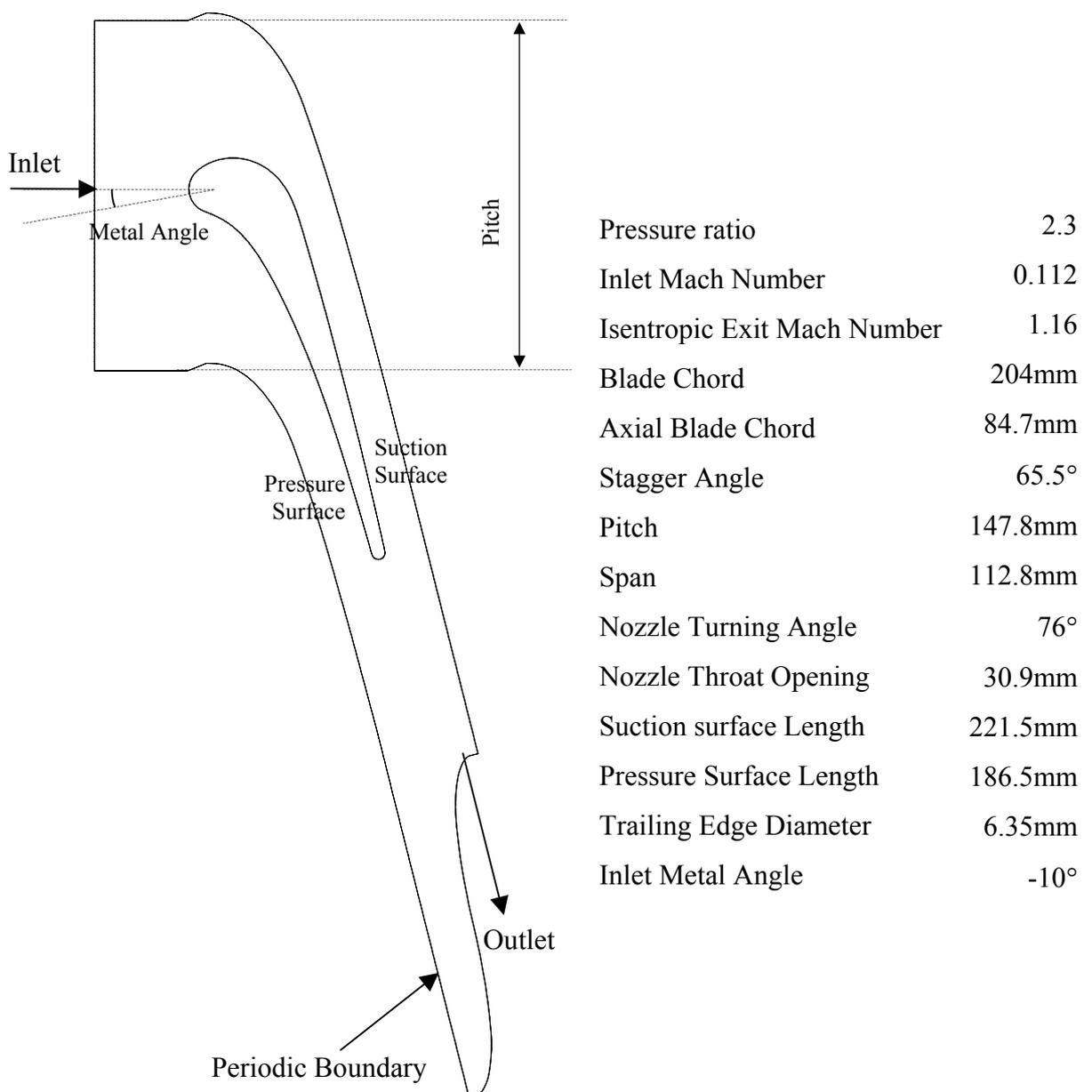
Figure 11.1: *Turbine Cascade Design Parameters*

This particular blade has already been used extensively in previous computational and experimental investigations due to its relatively thick trailing edge. For the current study, the flow conditions are defined by an inlet Mach number of 0.112, an exit Mach number of 1.16 and a turning angle of 76°; the flow in the trailing edge region is transonic. This corresponds to experimental work performed in Canada, Carscallen *et al.* (1998), measured a vortex shedding frequency in the order of 11-12kHz. At this frequency a detailed analysis of the structure of the wake is very difficult, making this a good candidate for alternative methods of study. In particular, the method of study offered is to model the blade computationally, and use the numerical data to design a larger scale experimental model.

As only the pressure distributions around the turbine blade are required at this stage, and not the unsteady phenomena in the wake region, the following computations are steady state using the implicit formulation defined in chapter seven.

The computational model is designed with upper and lower periodic boundaries. The non-adapted mesh can be seen in Figure 11.2, and the adapted mesh in Figure 11.3. The Mesh around the blade consists of a semi-structured region consisting of four layers. The mid-line between the first two layers defines the effective wall boundary of the blade. This can be seen in the inset of Figure 11.2. The structured layer ensures that the end-points of the four point stencils lie within the domain. See chapter 6.1. The periodic boundary is defined along the middle of the channel between two consecutive blades.

Figure 11.4 shows the Mach number contours. The upper and lower adjacent blades have been included to show the periodic nature of the solution. Shock waves can be clearly seen after the expansion fan at the trailing edge of the blade. Of particular interest is the reflection of the shock wave from the suction surface of the next lower blade in the row. It is particularly important that this is replicated in the flow around the water table model, as this shock wave will perturb the boundary layer, ultimately affecting the flow at the trailing edge. The pressure distribution around the blade can be seen in Figure 11.5, in comparison with recent experimental data, Table 11.1, produced by Jon Ackerman of the University of Leicester. The shock impingement can be seen at $x_{axial} = 0.71$. Agreement with the experimental data is excellent. Although a coincidence, pressure-tapping 19 can be seen to lie on the line of the shock. In the experimental case the shock would move very slightly and as such, the pressure tapping would measure a weighted average of the upstream and downstream pressures depending on how much the shock moved across the tapping. In this case the experimental measurement lies near the centre of the numerical shock. Upstream of the shock, a 'kink' can be seen in the pressure profile; this is attributed to the presence of a

separation bubble on the suction surface. The bubble has not formed in the computational work due to the low resolution of the grid in this region, but its existence could be predicted from the strong adverse pressure gradient at this point. Sanz & Platzer (1997) have produced similar results showing separation bubbles forming on the suction surface of compressor blades. Separation behind this particular blade has been verified by Moustapha *et al* (1993), over a range of Mach numbers including the current test case.

Tapping No.	x (mm)	y (mm)	$\dfrac{p}{p_\infty}$
1	-46.0602	48.5839	0.990672
2	-42.6732	40.54084	0.999111
3	-33.3646	35.91397	0.998108
4	-24.8973	29.97037	0.998098
5	-16.4336	19.69983	0.996624
6	-7.96991	0.489753	0.993618
7	0.497434	-13.2259	0.987074
8	6.84337	-28.7231	0.976987
9	13.19296	-46.1479	0.953784
10	32.23809	-108.729	0.496660
11	-46.0602	48.5839	0.991329
12	-42.6732	56.04906	0.946087
13	-33.3646	61.01243	0.921808
14	-20.6654	61.05632	0.792133
15	-12.2018	57.16829	0.682408
16	-3.73807	47.92919	0.614674
17	2.611526	31.83941	0.514840
18	8.96112	9.623146	0.373773
19	13.19296	-5.75706	0.344086
20	25.88849	-54.7945	0.437600
21	30.12399	-72.2413	0.415471
22	37.73912	-105.288	0.374461

Tapping Positions
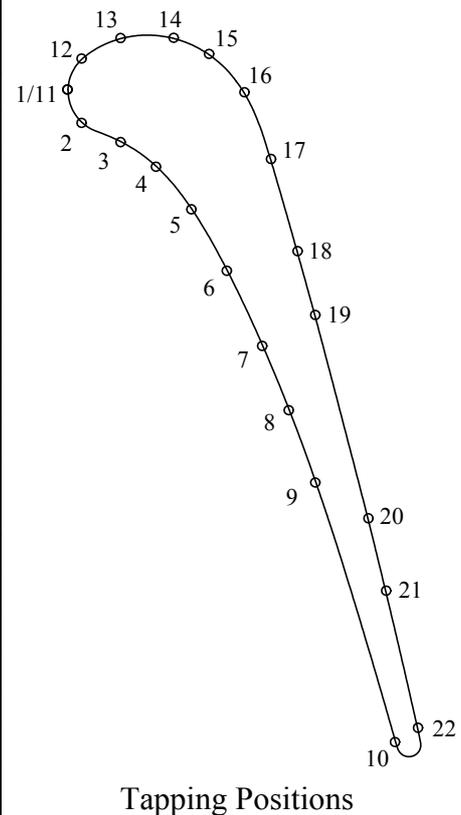
Table 11.1: *Experimental Pressure Measurements[15]*

11.1   Concluding Remarks

A computational study of the turbine blade under analysis has been presented in this chapter. It has been shown to be in excellent agreement with previous experimental work, and a good indicator of the accuracy of the numerical code.

---

[15] Courtesy: Jon Ackerman, University of Leicester. Taken at the CNRC, Ottawa, Ontario.
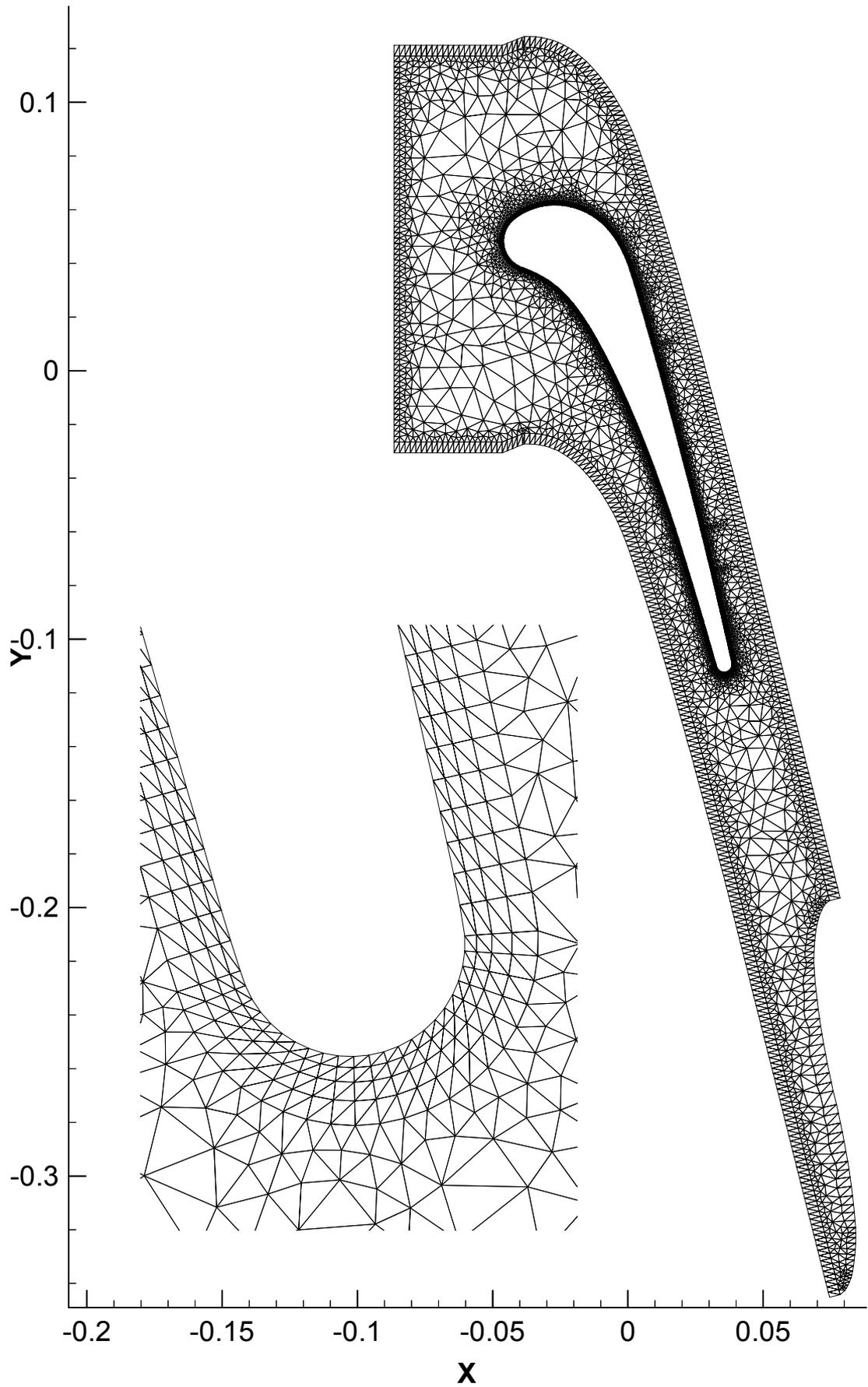
Figure 11.2: *Computational Mesh for Turbine Cascade*
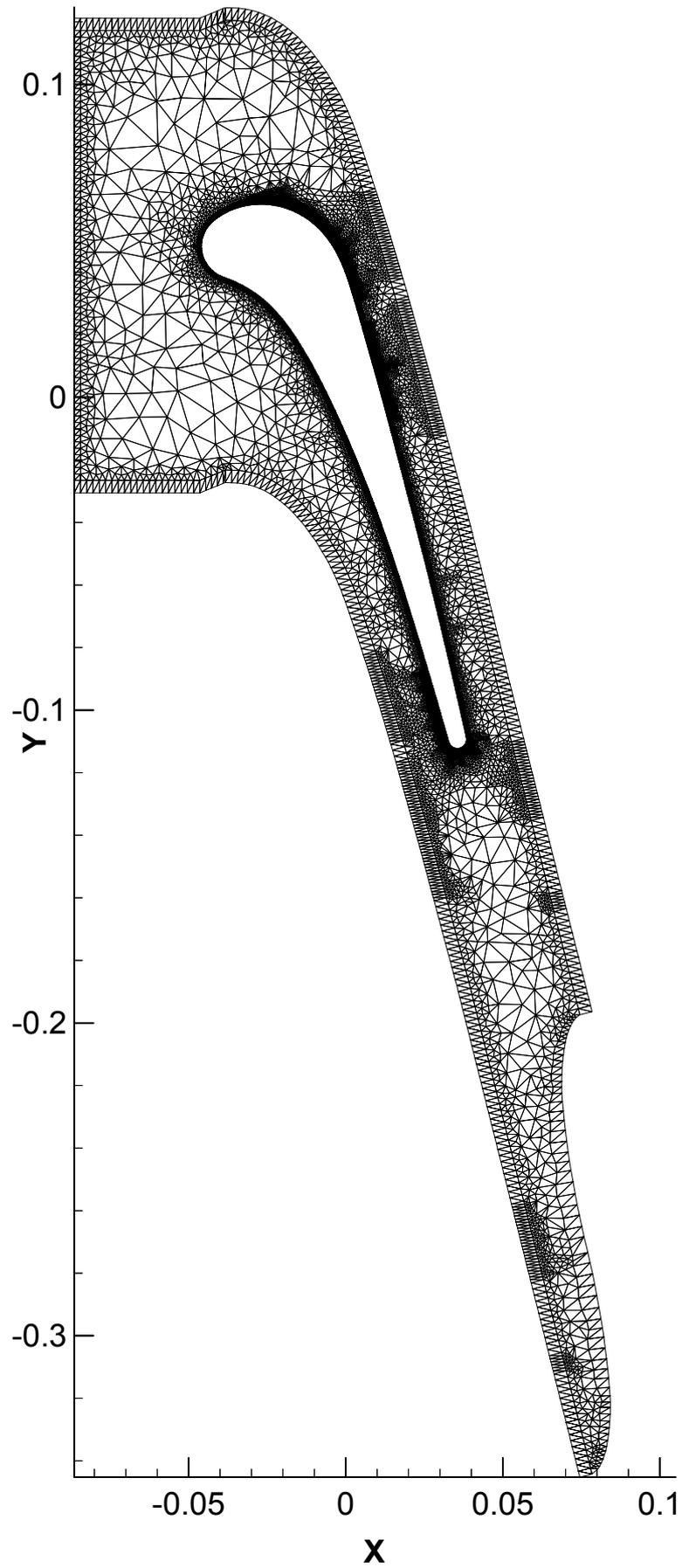
Figure 11.3: *Adapted Mesh for Turbine Cascade, 1 Adaptive Step*
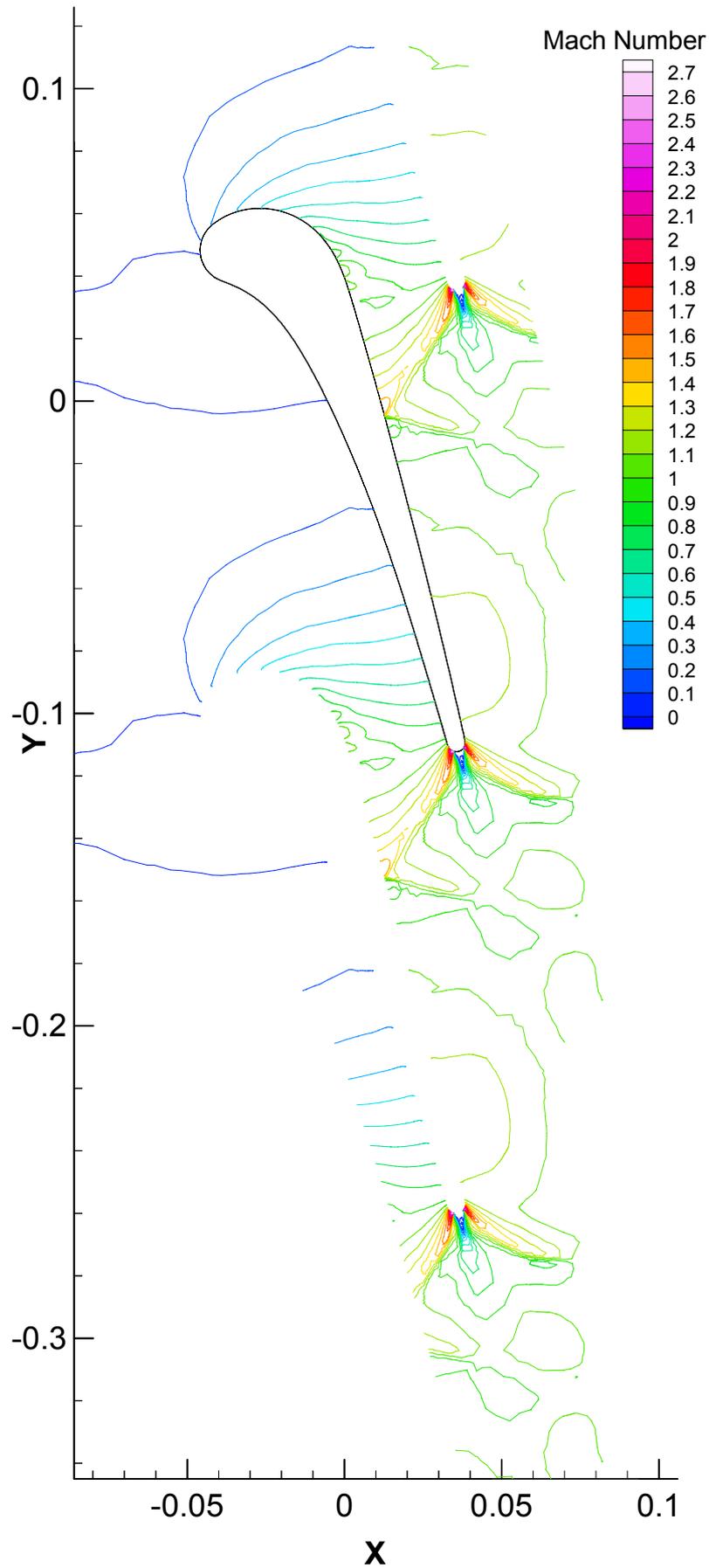
Figure 11.4: *Mach Number Contours for Adapted Turbine Cascade, ΔM = 0.1*
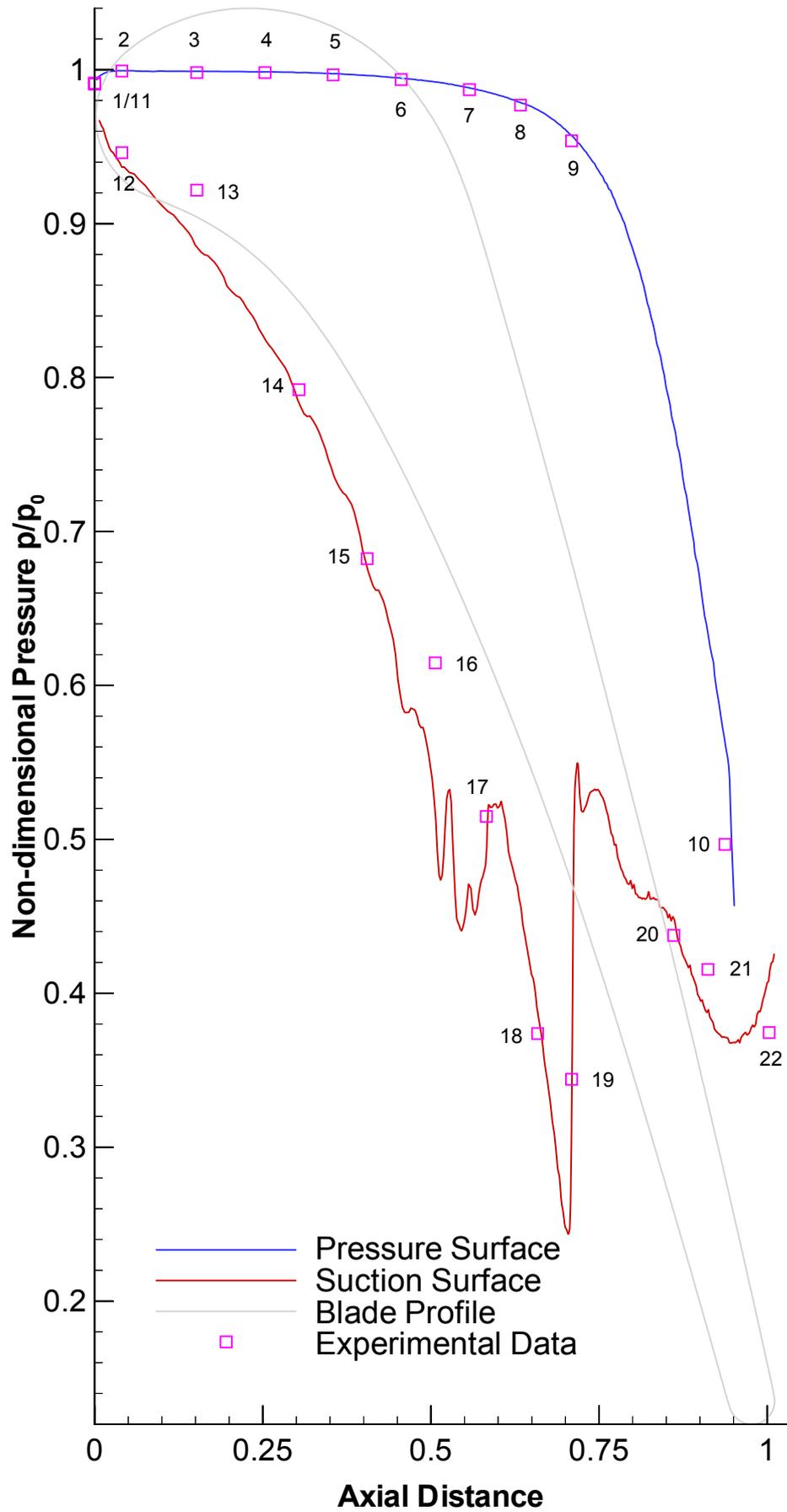
Figure 11.5: *Pressure Distribution for Turbine Cascade*

12   HYDRAULIC ANALOGY

An objective of the current study is to develop a water table model, based on the principle of open channel flow, that exhibits the same characteristics and flow phenomena that are associated with the turbine blade. The purpose being that experimental investigation of the turbine blade is problematic; the geometry of turbomachinery constrains the use of intrusive probes, and the velocity of transonic airflow requires the use of very high frequency data capture techniques. Carscallen (1998) measured vortex-shedding frequencies in the region of 11-12kHz for the turbine blade under study. If, for example, only ten samples are required between two consecutive vortices shed from the same surface of the blade, the sampling frequency would need to be in the order of 100kHz; this has only recently been achieved, Carscallen *et al* (1999). For a more detailed study, even higher frequencies are required. A water table model, however, has the advantage that for an equivalent model, transonic flow will occur above about 0.44ms^{-1} for a water depth of 0.02m, and vortex-shedding frequencies will be proportionately less. Visual analysis of flow phenomena may be achieved through simple techniques such as the use of video cameras and float gauges. Before this can be done, it is necessary to establish that the water table may indeed be used as an analogy of the air model. This is achieved through the concept of the hydraulic analogy; Shepherd (1965).

The hydraulic analogy depends upon the similarity between the equations describing two-dimensional compressible flow and those describing open channel flow with a free surface. Forces that are encountered in any fluid system may include inertia, viscosity, gravity, surface tension, pressure, and compressibility; Douglas *et al* (1995). A series of dimensionless constants can be used to describe the properties of any system in terms of these six forces:

$$M = \frac{U}{\sqrt{\dfrac{\gamma p}{\rho}}} \qquad\qquad \text{Mach number[16]} \qquad\qquad\qquad \textbf{(12.1)}$$

---

[16] Mach number: Ernst Mach (1838-1916). Austrian Physicist and Philosopher.

$$Re = \frac{\rho U L}{\mu} \qquad \text{Reynolds number}[17] \qquad (12.2)$$

$$Fr = \frac{U}{\sqrt{gL}} \qquad \text{Froude number}[18] \qquad (12.3)$$

$$We = \frac{U}{\sqrt{\dfrac{\sigma}{\rho L}}} \qquad \text{Weber number}[19] \qquad (12.4)$$

where U is the characteristic velocity, L the characteristic length, μ is the molecular viscosity, g the acceleration due to gravity, σ the surface tension and ρ the density. For complete similarity, based on two models surrounded by the same working fluid, all the above must match between the prototype and the model. For the current study, however, where similarity is to be achieved between a model in air and a water table model, similarity may be achieved by comparison of Mach number with Froude number. The flow is transonic; therefore both the systems are dominated by pressure waves.

12.1   The Hydraulic Jump

Consider water or any liquid flowing in an open channel; any small disturbance in the flow will be propagated along the channel at a velocity governed by the depth of the water; Graf (1998). If the height of the wave is sufficiently smaller than the depth of the channel then the velocity of propagation of the wave, also known as its *celerity*, will be:

$$c = \sqrt{gD} \qquad (12.5)$$

where D is the depth of the channel. If the velocity of the wave is the same as the velocity of the flow then the flow velocity can be expressed as $Fr = 1$ from (12.3). For $U < c$ the flow is expressed as *subcritical* and for $U > c$ the flow is *supercritical*. For supercritical upstream flow, a hydraulic jump will present itself. A typical hydraulic jump is shown in Figure

---

[17] Reynolds number: Osborne Reynolds (1842-1912). Professor of Engineering at Manchester University.

[18] Froude number: William Froude (1810-1879). Pioneer in the study of Naval Architecture.

[19] Weber number: Moritz Weber (1871-1951). German Naval Architect.

12.1(a); the jump is normally smeared over a finite width. Figure 12.1(b) shows the idealised instantaneous jump condition.

(a) Real Hydraulic Jump   **(b)** Ideal Hydraulic Jump
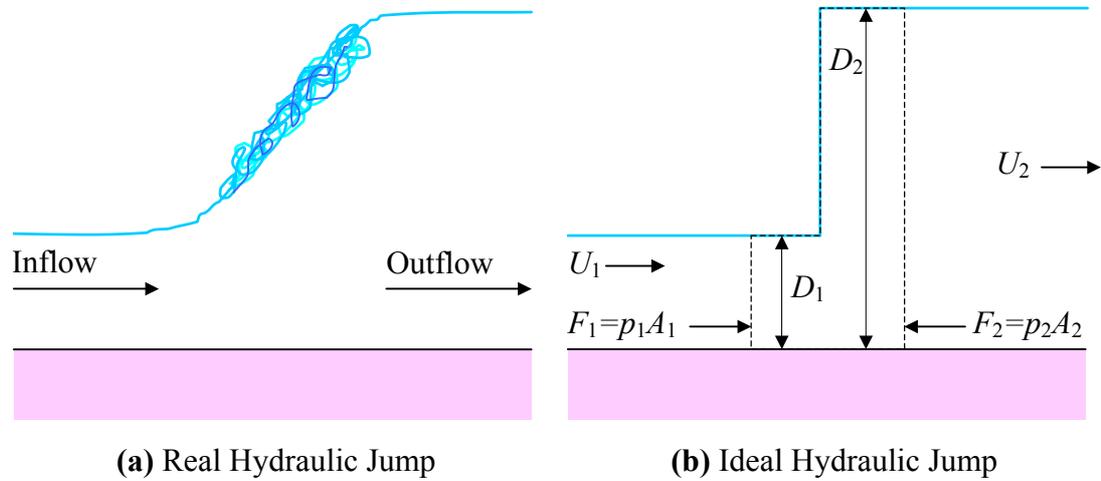
Figure 12.1: *Real and Ideal Hydraulic Jump*

This flow criterion can be seen as analogous to compressible gas flow. In this case the velocity of wave propagation is:

$$a = \sqrt{\frac{\gamma p}{\rho}} \qquad\qquad (12.6)$$

For $U > a$ the flow is considered to be *supersonic* and $U < a$ the flow is *subsonic*. From (12.1), a velocity of $U = a$ will give $M = 1$. One can clearly see that an analogy may exist between a hydraulic jump in channel flow and a plane normal shock in compressible flow. To verify this, the continuity, momentum and energy equations must be more closely examined.

12.2   Continuity

For a one-dimensional compressible flow, the continuity equation may be expressed as:

$$\dot{m} = \rho_1 U_1 A_1 = \rho_2 U_2 A_2 = \text{constant} \qquad\qquad (12.7)$$

where $\dot{m}$ is the mass flow rate, ρ is the density, U the mean velocity, and A the cross-sectional area of the flow. Thus, in its differential form for a uniform cross sectional area:

$$U\partial\rho + \rho\partial U = 0 \qquad\qquad (12.8)$$

For open channel flow, the continuity equation is given as:

$$\dot{m} = \rho_1 U_1 A_1 = \rho_1 U_1 B_1 D_1 = \rho_2 U_2 B_2 D_2 = \text{constant} \tag{12.9}$$

where U is the mean velocity, B is the channel width, and D is the channel depth. As the flow is incompressible, ρ is constant, and cancels out. In its differential form for a uniform channel width, as in the above case:

$$U\partial D + D\partial U = 0 \tag{12.10}$$

One can now see a clear comparison between (12.8) and (12.10) with the channel depth being analogous to density. This analogy can be extended to two dimensions without loss of generality.

12.3   Conservation of Momentum

With an instantaneous change over the hydraulic jump as in Figure 12.1(b), the only forces acting on the fluid are those due to pressure, thus the momentum equation for a channel of unit width reduces to:

$$p_1 D_1 - p_2 D_2 = \dot{m}(U_2 - U_1) \tag{12.11}$$

where $\dot{m}$ is the mass flow rate. As the particular forces in question are those due to hydrostatic pressure, p can be taken as the mean pressure at the centroid of the face:

$$p = \frac{1}{2}\rho g D \tag{12.12}$$

Making the above substitution, and using the continuity equation (12.9), the momentum equation becomes:

$$D_1^2 - D_2^2 = \frac{2U_1}{g}(U_2 - U_1) \tag{12.13}$$

which can be rearranged to give:

$$D_1^2 - D_2^2 = \frac{2U_1^2}{gD_2}(D_1 - D_2) \tag{12.14}$$

therefore

$$gD_2^2 + gD_1D_2 - 2U_1^2 = 0 \tag{12.15}$$

Solving for D_2 and dividing through by D_1 gives the ratio:

$$\frac{D_2}{D_1} = -\frac{1}{2} + \sqrt{\frac{1}{4} + \frac{2U_1^2}{gD_1}} \tag{12.16}$$

Examination of (12.16) shows that for flow velocity equal to the celerity then $D_2 = D_1$, and for supercritical flow, $D_2 > D_1$. The case for $D_2 < D_1$ will be investigated in the following text.

12.4   Conservation of Energy

The energy equation for channel flow is given by:

$$D_1 + \frac{U_1^2}{2g} = D_2 + \frac{U_2^2}{2g} + E_L \tag{12.17}$$

The energy losses E_L would be those caused by shear stresses due to surface tension. If the downstream condition is taken to be very deep and the energy losses very small, then the velocity at this point can be assumed to be negligible, representing stagnation conditions:

$$D_1 + \frac{U_1^2}{2g} = D_s \tag{12.18}$$

where D_s is the stagnation depth. This then gives a maximum velocity, as D_1 tends to zero:

$$U_{max}^2 = 2gD_s \tag{12.19}$$

The velocity for a given depth can then be derived from:

$$\left(\frac{U}{U_{max}}\right)^2 = \frac{D_s - D}{D_s} \tag{12.20}$$

For compressible flow, assuming there are no energy losses, the energy equation is:

$$\frac{p_1}{\rho_1} + \frac{U_1^2}{2} + u_1 = \frac{p_2}{\rho_2} + \frac{U_2^2}{2g} + u_2 \tag{12.21}$$

Thus by introducing the thermodynamic relations:

$$u = C_v T \qquad C_p - C_v = R \qquad p = \rho R T \qquad \text{and} \qquad \gamma = \frac{C_p}{C_v} \tag{12.22}$$

where u is the internal energy, C_v and C_p are the specific heat constants for constant volume and pressure respectively, T is the temperature, R is the universal gas constant, and γ the ratio of specific heats; the energy equation reduces to:

$$\frac{\gamma}{(\gamma - 1)} \frac{p}{\rho} + \frac{U^2}{2} = \text{constant} \tag{12.23}$$

Substituting for the velocity of wave propagation a into the above gives:

$$\frac{a^2}{(\gamma - 1)} + \frac{U^2}{2} = \text{constant} \tag{12.24}$$

As with the channel flow, the stagnation and maximum conditions can be derived as:

$$\frac{a_s^2}{(\gamma - 1)} = \text{constant} \qquad \frac{U_{max}^2}{2} = \text{constant} \tag{12.25}$$

and the velocity can then be expressed as:

$$\left(\frac{U}{U_{max}}\right)^2 = \frac{a_s^2 - a^2}{a_s^2} \tag{12.26}$$

For similarity of velocities, a comparison of equations (12.20) and (12.26) gives:

$$\frac{a^2}{a_s^2} = \frac{D}{D_s} \tag{12.27}$$

but since $a = \sqrt{\dfrac{\gamma p}{\rho}}$ this can be expressed as:

$$\frac{a^2}{a_s^2} = \frac{\rho_s}{\rho} \cdot \frac{p}{p_s} \qquad\qquad\qquad\qquad (12.28)$$

Now the hydraulic analogy relies on the analogy between depth in channel flow and density in compressible flow. For this to be satisfied, the following relation must hold:

$$\frac{D}{D_s} = \frac{\rho}{\rho_s} \qquad\qquad\qquad\qquad (12.29)$$

This can be satisfied from (12.27) and (12.28) if

$$\frac{\rho}{\rho_s} = \frac{\rho_s}{\rho} \cdot \frac{p}{p_s} \qquad\Rightarrow\qquad \frac{p}{\rho^2} = \text{constant} \qquad (12.30)$$

but for adiabatic flow $\dfrac{p}{\rho^{\gamma}} = \text{constant}$, thus the water table flow becomes analogous to a gas with ratio of specific heats $\gamma = 2$. This is representative of the operating conditions of a steam turbine running at a temperature of 700K and a pressure of 20-25MPa[20]. The hydraulic analogy would therefore be highly accurate for modelling such a case.

12.5   Energy Losses

By taking into account the energy loss term E_L in (12.17), across the hydraulic jump this can be interpreted as a loss in mechanical energy.

$$E_L = D_1 - D_2 + \frac{U_1^2 - U_2^2}{2g} \qquad\qquad\qquad (12.31)$$

Substituting for the continuity equation ($U_1 D_1 = U_2 D_2 = \text{constant}$) into the above:

---

[20] Obtained from NIST (National Institute of Standards and Technology) website; http://www.nist.gov

$$E_L = \frac{(D_2 - D_1)^3}{4D_1 D_2}$$

           (12.32)

By definition, E_L must be positive, as there can be no increase in mechanical energy; this would violate the second law of thermodynamics. Therefore D_2 will be greater than D_1 implying from (12.16) that U_1 must be supercritical. Furthermore, it may also be noted that as $D_2 \to D_1$, then E_L tends to zero and the hydraulic jump tends to the ideal. For the equivalent compressible flow, E_L this will manifest itself as a loss in stagnation pressure.

From the above analysis, it can be shown that there is a close qualitative analogy between the hydraulic jump and a normal shock wave, see Table 12.1:

Hydraulic Jump	Normal Shock
Celerity $c = \sqrt{gD}$	Wave speed $a = \sqrt{\dfrac{\gamma p}{\rho}}$
Subcritical flow	Subsonic flow
Supercritical flow	Supersonic flow
Possible only for $U_1 > c_1$ and $U_2 < c_2$	Possible only for $U_1 > a_1$ and $U_2 < a_2$
Loss of mechanical energy	Loss of stagnation pressure

Table 12.1: *Comparison of Hydraulic Jump and Normal Shock*

The above calculations have been based on the assumption that there are no energy losses, though this is clearly invalid as there are mechanical losses and loss of static pressure in hydraulic jumps and shock waves. However, the analogy may be used with minimal losses if the flow is restricted to the subsonic to transonic flow region.

12.6   Reynolds Number

In the case where both the prototype and the model are assumed to be inviscid, the effects of viscosity are assumed to be negligible, consequently the Reynolds number can be ignored in both cases. It must be noted however, that the experimental model has viscous effects

associated with it. In this case, Reynolds number must be accounted for, particularly when considering unsteady phenomena such as vortex shedding.

The conditions for the turbine blade are:

$$
\begin{bmatrix} M \\ P_0 \\ T_0 \\ L \\ \mu \end{bmatrix}_{Blade} = \begin{bmatrix} 0.112 \\ 101352 \\ 293 \\ 0.204 \\ 17.05 \times 10^{-6} \end{bmatrix}
\tag{12.33}
$$

giving a calculated Reynolds number, based on chord length, of:

$$
Re_{Blade} = 507,096
\tag{12.34}
$$

The computational water table model is a one-to-one scale model of the physical model, so the Reynolds numbers will match. For the actual water table model:

$$
\begin{bmatrix} Fr \\ \rho \\ L \\ \mu \end{bmatrix}_{WT} = \begin{bmatrix} 0.112 \\ 10^3 \\ 0.9135 \\ 10^{-3} \end{bmatrix}
\tag{12.35}
$$

For matching Reynolds numbers, Re_{WT} must equal Re_{Blade}.

$$
Re_{Blade} = Re_{WT} = \frac{10^3 . U_{WT} . 0.9135}{10^{-3}}
\tag{12.36}
$$

to give the inlet velocity of the water table to be $U_{WT} = 0.55 \text{ms}^{-1}$. Unfortunately, for this to be equivalent to a Mach number of 0.112, the required depth of the inlet must be 2.5 metres, clearly impractical for the current study; the flume at the university is only 30cm deep. Based on this information, it is appropriate to find a balance between the Reynolds numbers calculated and the applicability of the hydraulic analogy. The water table geometry that will be described in the following chapter has been designed to fill the flume at the University of Leicester; the larger the model, the closer the Reynolds number will match with the turbine blade.

In order for the hydraulic analogy to hold, the depth of the water must be restricted to minimise three-dimensional effects. It is reasonable to hold the inlet depth at about 0.02-0.025m, this gives an inlet wave speed of 0.443-0.495ms^{-1}. For a Mach number of 0.112, this corresponds to an inlet velocity of 0.0496-0.0555ms^{-1}, and a Reynolds number of only 50,356.

The condition of the boundary layers is very important when considering Reynolds number matching as small changes in Reynolds number can alter the state of a boundary layer from laminar to turbulent. For the current study, the pressure surface of the turbine blade can be assumed to be laminar due to the lack of substantial diffusion on that surface. The suction surface contains various states: from the leading edge the boundary layer is laminar, the boundary layer separates and re-attaches further downstream, and the impingement of the shock triggers a turbulent boundary layer to the trailing edge of the blade. Between the re-attachment point and the shock, the boundary layer may be transitional. For the water table model, the separation bubble is not modelled; consequently the conditions for this section of the suction surface may be inaccurately resolved. Fortunately, the presence of the shock impingement ensures that the probability of the suction surface boundary layer being in the correct turbulent state at the trailing edge is very high. Further steps to address this may be taken by modifying the surface roughness of the flat plate, or introducing boundary layer trips to promote the correct boundary layer state.

12.7   Weber Number

To ensure that vortex shedding is accurately portrayed by the hydraulic analogy it is also necessary to examine the importance of Weber number. Considering the Euler equations appropriate to a compressible gas flow, there would be no contribution from surface tension, and the Weber number is unimportant. However, the Weber number is important in the comparison of the numerical water table simulations with the experimental water table. Surface tension is now present and it is necessary to ensure that the surface tension can either be ignored or accounted for by the viscous terms of the numerical method.

Surface tension is produced because all molecules on average are attracted equally to the surrounding molecules. At the interface between air and water, a molecule has a net force attracting it into the bulk of the fluid. This has the effect that the fluid acts as if a membrane under tension were present at the interface. The overall result of this is the minimising of the surface area. In the case of the water table, this would cause smearing of the shock waves,

smoothing of vortices in the trailing edge region, and effects along the boundaries and the base of the model.

In the current study, the effects of surface tension have been ignored, as this would require the addition of extra terms to the Navier-Stokes code. However, careful attention should be paid to the smoothing effects to ensure that they do not invalidate the comparison.

12.8   Vorticity

The vorticity of a fluid element within a flow, Douglas *et al* (1995), is defined by the equation:

$$\Omega = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$$
(12.37)

and is a measure of the degree of rotation of that element. Vorticity is applicable to both gas flow and water table flow as it is only a function of velocity. The velocity has been shown to be analogous in the above text.

12.9   Concluding Remarks

The hydraulic analogy has been presented in this chapter. Through this analogy, two-dimensional flow around turbomachinery can be accurately simulated by open channel flow. The application of the analogy simply consists of a shallow sheet of water flowing over an almost horizontal table. The slight gradient overcomes the resistance due to the contact with the surface of the water-table. Two-dimensional representations of the prototype to be tested are then placed on the table. Visualisation techniques such as shadowgraph, video cameras, and float gauges can be used for which a great deal of literature is available.

It has been shown that, neglecting viscous and surface tension effects, the Euler equations can be used for the study of flow around turbomachinery blades. In this manner, density is analogous to the depth of flow in the water table. Inaccuracies will be present, however, when the remaining viscous terms and surface tension are taken in account. For the current study, no further matching of Reynolds number can be achieved by the methods described earlier, and surface tension will cause smearing of the shock waves and damping of the vortices.

Several techniques can be used to improve the characteristics of the physical model by seeking to reduce the viscosity and minimise the surface tension.

1)   Increasing the water temperature will have the effect of decreasing the kinematic viscosity; the combined effect will be an increase in the Reynolds number to match the turbine blade. The effects of temperature on the required depth, and the water table Reynolds number if the inlet is held at 0.025m deep, can be seen in Table 12.2. This shows that it may be feasible to minimise the effects of the difference in Reynolds number between the prototype and the model. The temperature increase would also have the additional effect of reducing the surface tension.

2)   The addition of surfactant molecules to the water would have to effect of reducing the surface tension. Surfactants have not been investigated in the current study, but will be important in further experimental work. The flume used for the current study is fed by a continuous source from a large recycled reservoir. In this case, the use of a surfactant would be undesirable as continuous injection upstream of the model would be required. This would eventually contaminate the reservoir that is used for several other projects. An independent, small-scale reservoir would be needed for which the concentration levels of the surfactant can be accurately controlled.

3)   It may also be possible to investigate other fluids with lower kinematic viscosity. These would give a higher Reynolds number for the water table, giving a better match with the turbine blade.

Temperature (°C)	Kinematic Viscosity ($m^2 \, s^{-1}$)	Surface Tension (Nm^{-1})	Re_{WT}	Required Inlet depth (m)
20°	1.007 ($\times 10^{-6}$)	7.36 ($\times 10^{-3}$)	50356.21030	2.538
40°	0.661	7.01	76682.61563	1.095
60°	0.477	6.88	106284.6537	0.57
80°	0.367	6.3	138009.8912	0.338
100°	0.296	5.94	171092.1169	0.22

Table 12.2: *Required Inlet depths*

13   WATER TABLE GEOMETRY

Two sets of results have been computed for the water table geometry. The first set comprises the computations required to generate a comparable Mach number distribution to the turbine blade geometry. The second set of results is a time dependent simulation to generate an unsteady flow field, with a view to capturing unsteady phenomena. The initial grid is sufficient to generate the steady pressure distributions, as with the turbine blade. However, for the unsteady case, a denser grid is required in the trailing edge region. The purpose of the denser grid is to give improved resolution of the unsteady flow phenomena. The water table computations were performed with the same inlet conditions as the turbine blade, with the exception that a γ of 2 was chosen for compatibility with the hydraulic analogy.

13.1   Mach Number Distribution

To generate a geometry for the water table, an initial simple grid was first constructed, this was very similar to that given in Figure 13.1. Several modifications to the grid were necessary before reaching its current state. This was important, if particular features of the flow around the Turbine Blade were to be taken into account. The most important feature is the impingement of the trailing edge shock of an adjacent blade onto the suction surface of the current blade.
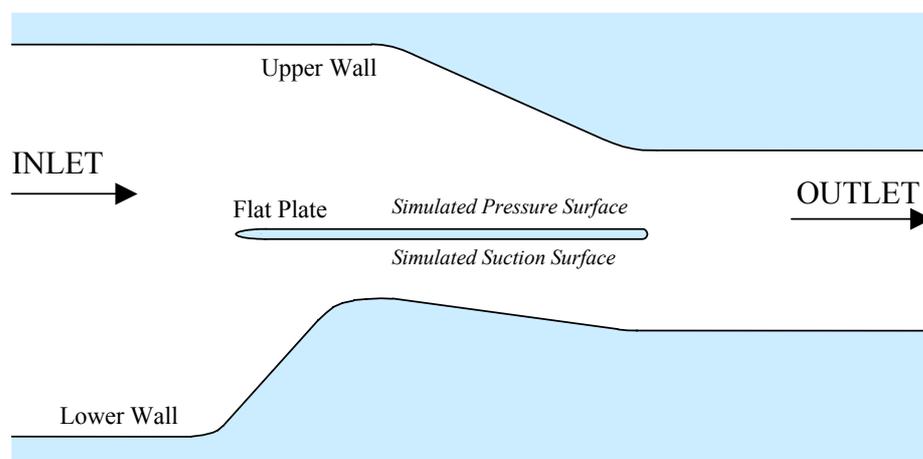
Figure 13.1: *Water Table Geometry*

A schematic of the water table geometry can be seen in Figure 13.1. The model is designed to given a Mach number distribution around the flat plate that is equivalent to the Mach number distribution around the turbine cascade; this is based on a similar model by Roberts & Denton (1996), designed for subsonic flow. The flat plate dimensions are: *height* = 25.4mm, *length* = 912.375mm, *trailing edge diameter* = 25.4mm, *leading edge* = 4:1 ratio ellipse to match the height of the blade. The current dimensions were chosen such that the model would fit the water table flume in the University of Leicester Hydrodynamics Laboratory, thereby giving the largest Reynolds number possible.

The upper surface of the flat plate is designed to simulate the pressure surface of the turbine blade, and the lower surface the suction surface of the turbine blade. With regard to the computational mesh, there are four layers at the surface of the flat plate, as with the turbine blade. The outer wall boundaries were created through an iterative process from an initial crude approximation very similar to Figure 13.1. The process involved manually inspecting the Mach number distribution around the flat plate and continuously making modifications to the geometry until the distributions, Figure 13.7, for the flat plate and the turbine blade were considered to be in good agreement. At this point, the main features of the Mach number distribution had been captured and further modification was deemed impractical.

The non-adapted mesh is given in Figure 13.2, and the adapted mesh in 13.3. The adapted mesh contains 79,384 points. The Mach number contours for the adapted mesh are given in Figure 13.4. A qualitative analysis of the Mach number contours shows that a similar expansion fan and shock structure is present downstream of the trailing edge of the blade. A shock wave impinging on the suction surface is also evident. This has been achieved by a small converging diverging section of the outer wall at that point, and can be seen more clearly in Figure 13.5. The Mach number distribution can be seen in Figure 13.6 with the profile for the flat plate. Figure 13.7 shows the comparison between the Mach number profiles for the turbine cascade and the water table geometry.

A close agreement can be seen between the two with the main features being present in both. The shock impinging on the suction surface of the water table is clearly visible and of a comparable magnitude to the turbine blade, although slightly more upstream. This displacement is not considered a problem, but could be rectified in further work.

There are, however, some distinct differences between the water table distribution and the turbine blade distribution:

1)   Upstream of the shock wave, the flow is smooth on the suction surface; this is does not correctly match the turbine blade in which separation occurs at this point, and can be seen as the increase in Mach number. The fact that separation is not present in the water table model is due to the fact that the pressure gradient is more severe in that location of the turbine blade model.

2)   The suction surface distribution is in general of the correct shape, but displaced upwards on the graph.

3)   The pressure surface distribution matches exactly at the trailing edge but is too low at the leading edge.

4)   At the leading edge of the flat plate, the angle of incidence of the flow can be seen to be aligned with the plate geometry; this can be deduced from the matching peaks in the pressure and suction surface pressure distributions. This does not correspond with the turbine blade behaviour, as this has an inlet flow angle of $-10°$. This is not considered to be a serious problem since the investigation is focused on trailing edge flows, but may be corrected when more important differences have been addressed.

The discrepancies in Figure 13.7 are mostly due to a higher loading at both the leading and trailing edges of the cascade as compared to the water table model. A further development of this work might be to increase the loading on the water table model.

For points two and three; the flow in the water table is not subject to any turning effects that are present in the turbine cascade – the flow is simply from left to right, and there is no angle of incidence of the incoming flow. The absence of curvature effects may lead to differences in the characteristics of the boundary layers of the turbine blade and the water table model. A closer match would be obtained with further modifications to the geometry. The suction surface distribution is of the correct shape, implying that modifications to the corresponding wall would not be required. In order to simulate higher loading, it would be necessary to divert more flow through to pressure surface channel by expanding the walls in the upstream region. As a consequence, flow velocity would increase in the lower channel causing a decrease in the local pressure, thereby increasing the loading on the blade.

It should be noted that the pressure distributions obtained from a turbine cascade are a property of the series of blades acting interactively, a property that is absent from the water table geometry.
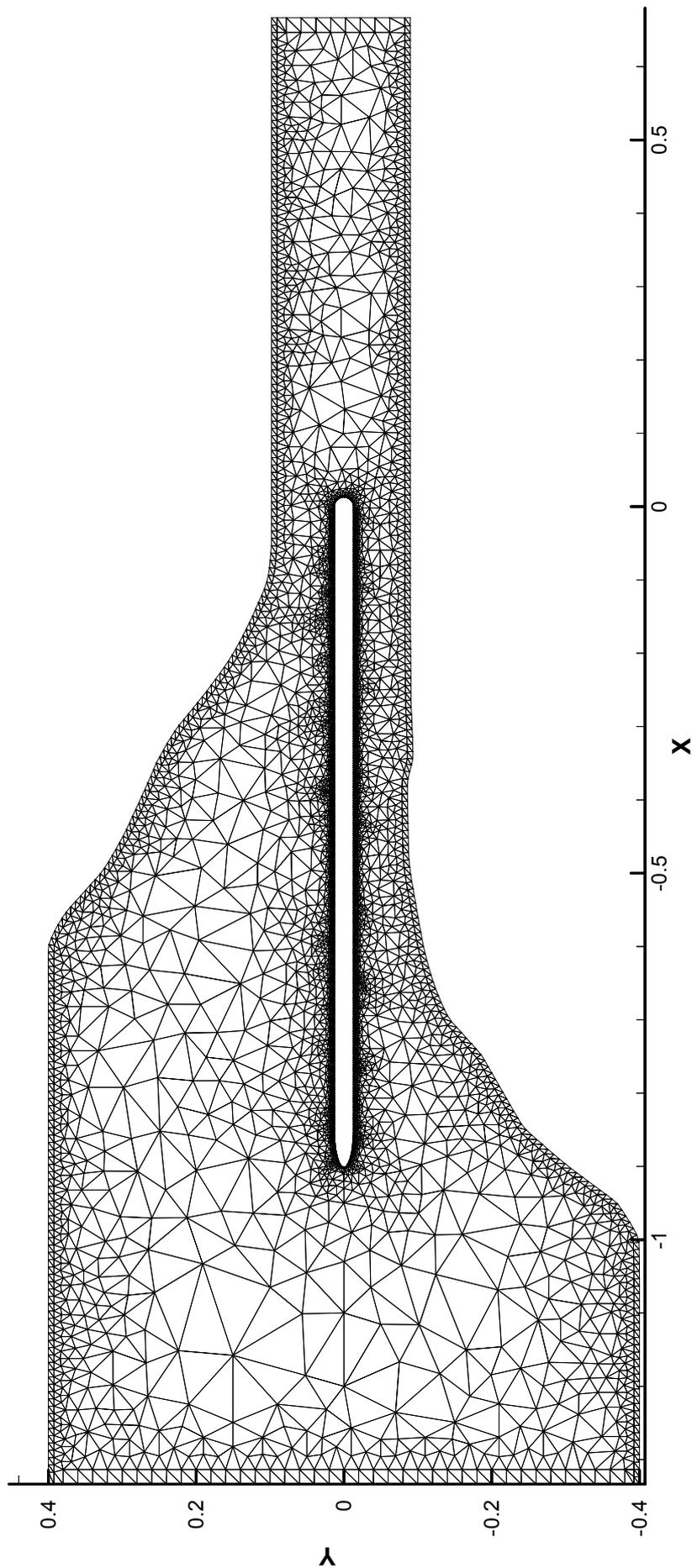
Figure 13.2: *Computational Mesh for Water Table*
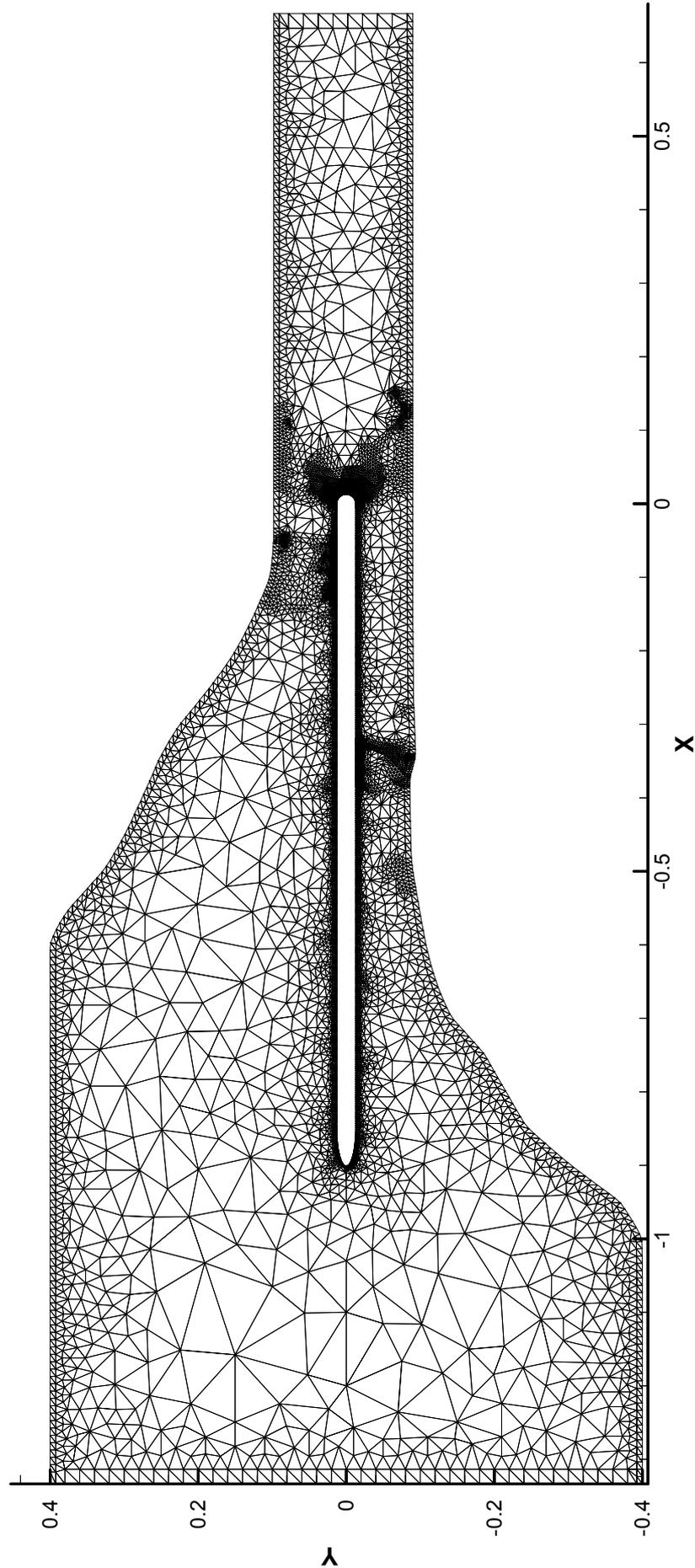
Figure 13.3: *Adapted Mesh for Water Table, 2 Adaptive Steps*
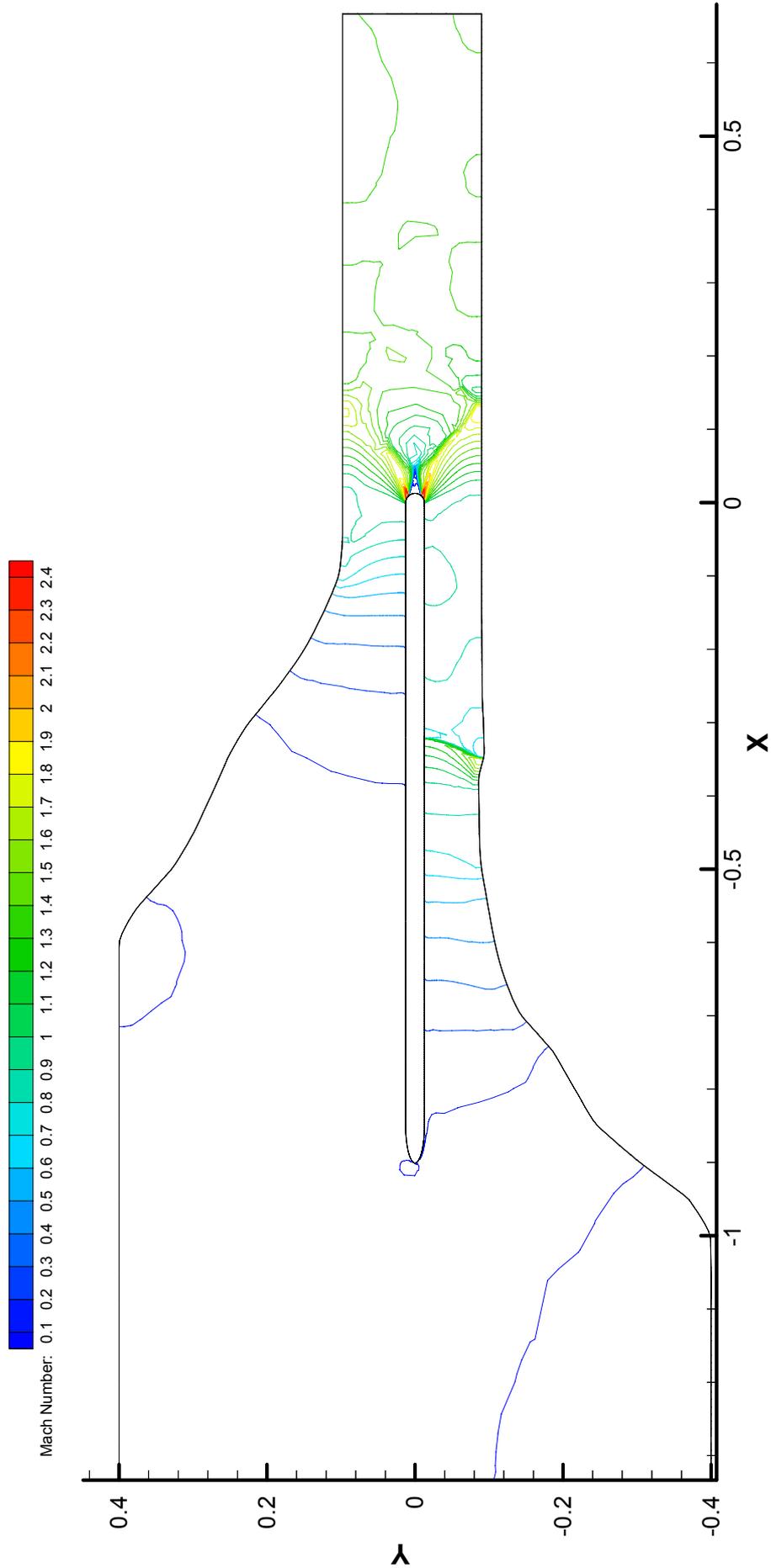
Figure 13.4: *Mach Number Contours for Adapted Water Table, ΔM = 0.1*
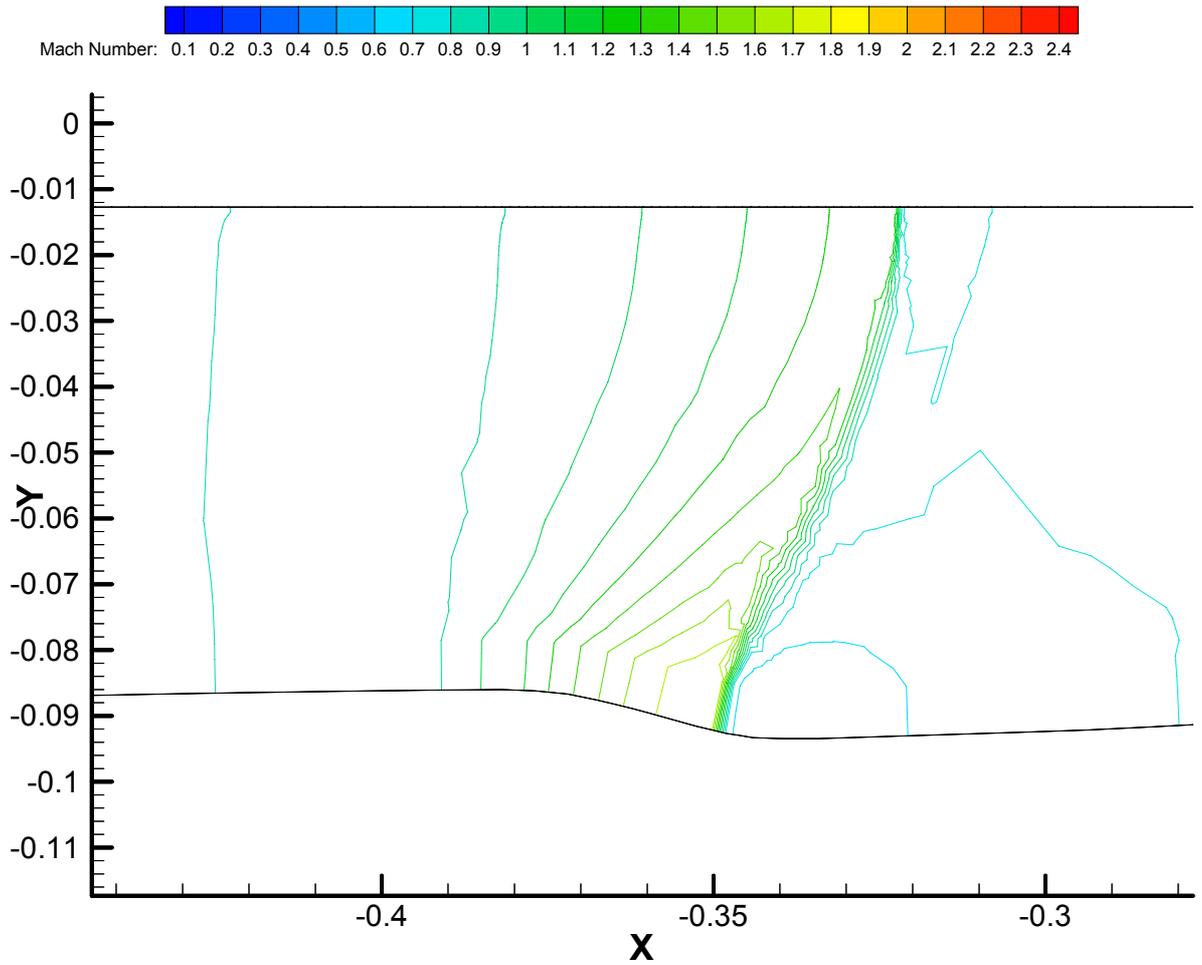
Figure 13.5: *Impinging Shock on Suction Surface*
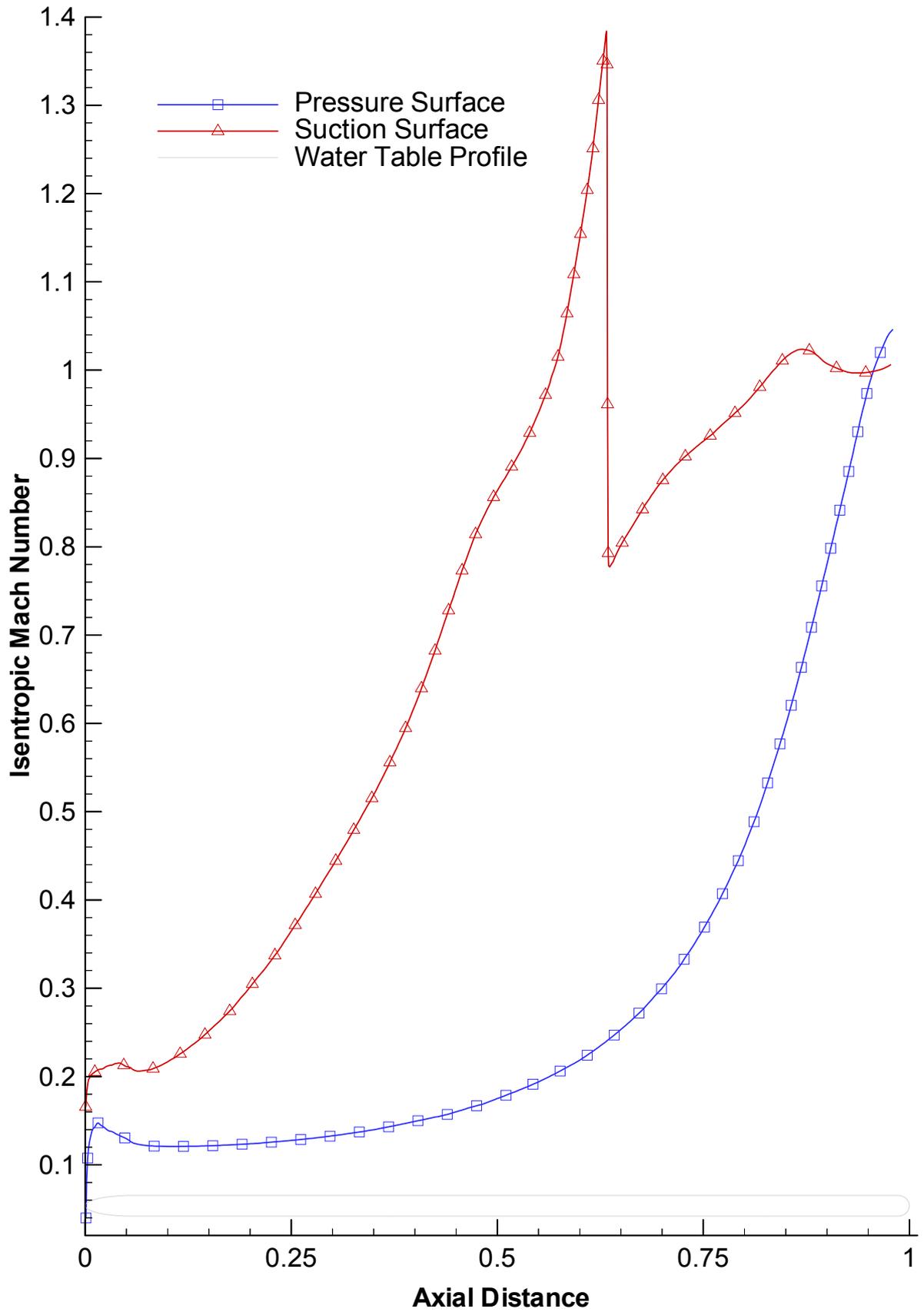
Figure 13.6: *Mach Number Distribution for Adapted Water Table*
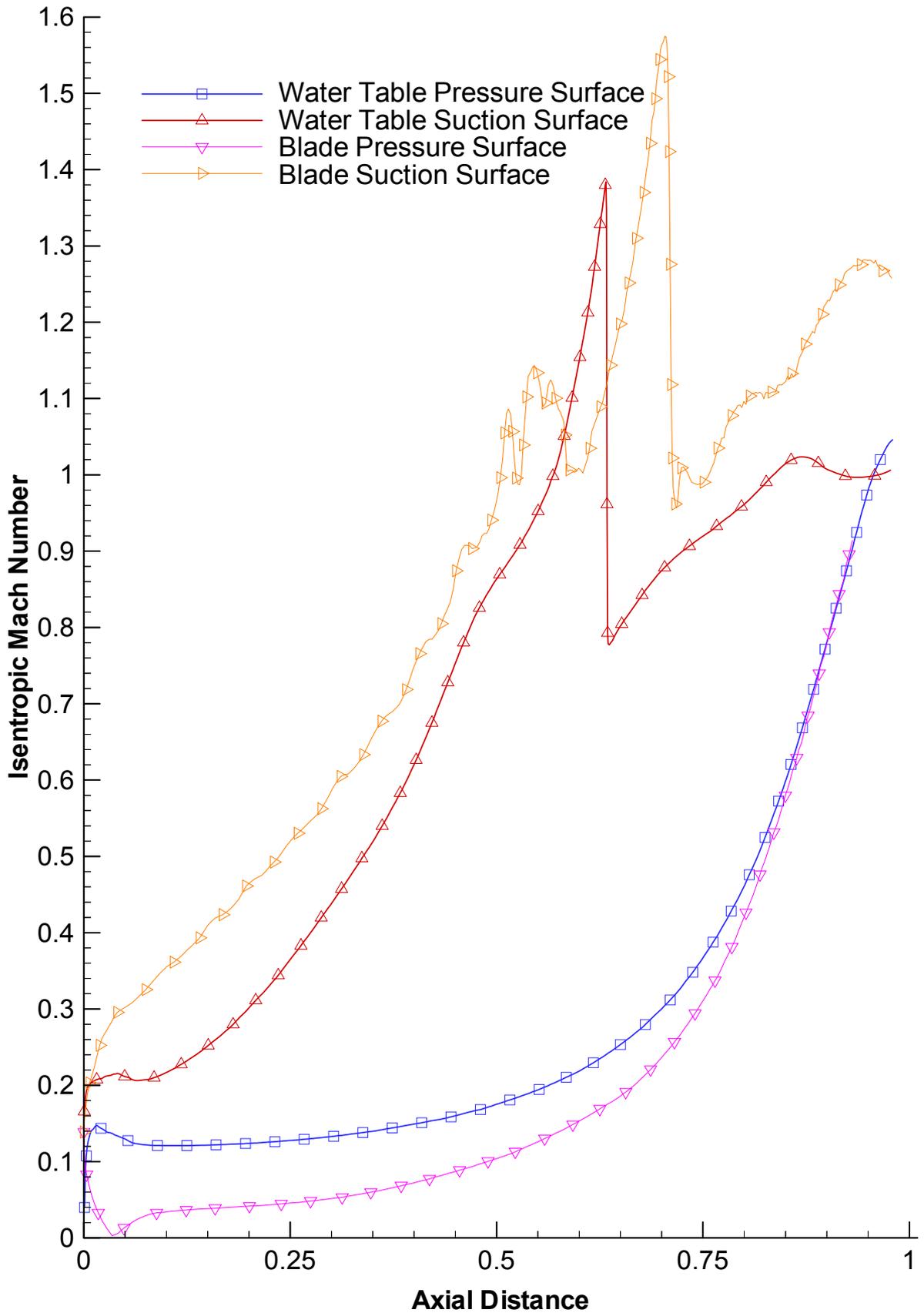
Figure 13.7: *Comparison of Blade and Water Table Mach Number Distributions*

13.2   Time Dependent Computation

To capture the unsteady phenomena, a slightly different gridding strategy is required for the time dependent study of the flow around the water table geometry. There is a great deal of activity in the trailing edge region of the flat plate, the point from which the vortex shedding originates. The dense grid for this region is given in Figure 13.9. The grid has been adapted to be at its most dense at the confluence region of the shear layers, at about $x = 0.06$. Grid adaption was performed manually over a series of rectangular zones; grid adaption based on density gradients would be impractical as the density gradients would be too low to active the procedure. There are now fifteen layers close to the surface of the plate. This is to accommodate turbulence modelling for future study, and to minimise any grid dependencies that may occur near the trailing edge.

The flow enters from the left hand side of the domain, on both the pressure and suction surfaces of the flat plate. A close-up of density contours in this region is given in Figure 13.10. As the flow rounds the curvature at the trailing edge an expansion fan forms. At about 40° around the edge, a shear layer is formed as the flow meets the re-circulation in the base region. The flow is straightened slightly, causing a compression shock to form. Kelvin-Helmholtz instabilities can be seen forming as a corrugation along the upper shear layer, a consequence of density and velocity gradients normal to the flow direction; Faber (1995). These propagate downstream to the confluence region of the upper and lower shear layers. A vortex structure can be seen emerging from the confluence region and spanning downstream. The Kelvin-Helmholtz instabilities can be seen more clearly in the enthalpy plot; Figure 13.11. These are more predominant in the upper layer corresponding to the pressure surface, which experiences a larger velocity and density gradient.

The instabilities can clearly be seen in the sequence of six density plots; Figure 13.12 to Figure 13.17. Additionally, these instabilities disturb the flow and cause smaller shock waves to be created, Figure 13.10. Above and below the confluence point, a pair of oscillating shocks are formed due to the turning of the flow to the outlet direction. These are created at the leading edge of the newly formed vortices emerging from the base region. The compression shocks from the trailing edge and the above mentioned shocks then merge to form the recognised 'fish-tail' shock structure. These are pointed out in Figure 13.10, and can be seen more readily in full-scale contour plot, Figure 13.4.

The flow entering the left hand side of the picture is roughly sonic. As the flow rounds the curvature of the blade, the Mach number increases to supersonic. After the shock structure,

the flow is temporarily subsonic and quickly becomes supersonic again. In the triangular base region the flow is subsonic. The shear layers defining the upper and lower bounds of this region are at the subsonic/supersonic interface. The centreline of the wake is subsonic.

The author posits the following phenomena interacting in the trailing edge region. In subsonic flow, the effective trailing edge is the trailing edge of the plate, Figure 13.8(a). Vortex shedding stems from the boundary layer separation points and the familiar von Karman vortex street is observed. As the flow becomes transonic and then supersonic, the flow acts as described in the previous paragraph. The effective trailing edge of the flat plate now looks more like that seen in Figure 13.8(b). Vortex shedding now stems from the smaller effective trailing edge resulting in higher frequency, lower energy vortices. If the trailing edge were thus shaped, a conventional von Karman vortex street would be observed. However, the situation now arises that the upper and lower surfaces of the extended section are dynamic. As the flow changes from type (a) to type (b), the base region initially has a very minor shear layer. Any instability in this layer is relatively small compared to the vortex shedding. As the Mach number increases, the effective trailing edge elongates. The vortex shedding decreases in magnitude and the instabilities in the shear layer become more dominant.

(a) Effective Subsonic Trailing Edge   **(b)** Effective Supersonic Trailing Edge
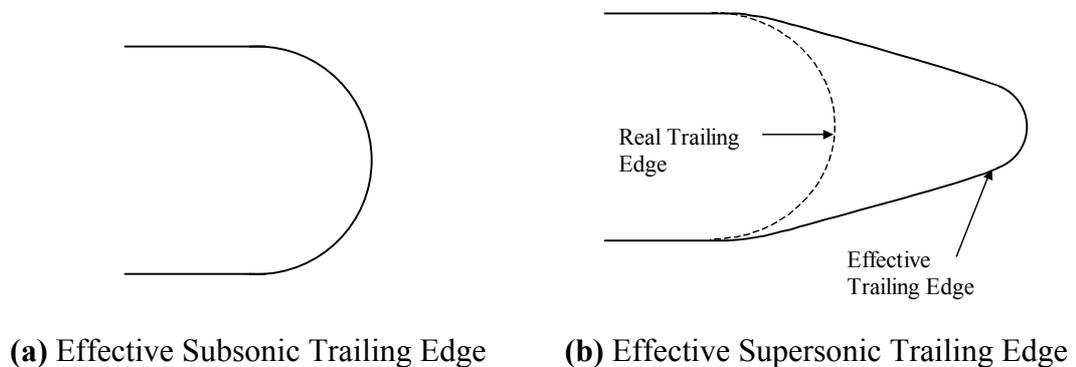
Figure 13.8: *Effective Trailing Edges Shapes*

There are two hypotheses at this point:

1)   It is the Kelvin-Helmholtz instabilities themselves that are squeezed through the confluence region to form the transient vortex shedding phenomenon. This raises the question of the relative length scales of the instabilities before and after the confluence region. The squeezing may be due to the 'shock waves' forming above the instabilities; each shock wave acting as a high-pressure barrier against expansion. There are no such structures in the wake region; the pressure gradients here are not large, with the pressure decreasing downstream.

2)   The Kelvin-Helmholtz instabilities interact with the already-present von Karman vortex street; it is thought that there is a point at which the instabilities are of such a magnitude that they disrupt the flow reaching the effective trailing edge, and as a consequence the vortex street collapses into the transient features described in the introduction. The wavelength of the von Karman vortex street is defined by the 'throat' of the confluence region, and the wavelength of the Kelvin-Helmholtz instabilities is determined by the viscosity and the velocity gradient across the shear layer. The frequency of the latter may affect the vortex shedding by causing the former to try to 'lock-in' to the new frequency. This may not be a natural mode for the vortex shedding, causing an unstable feed-back loop. As a result, transient features present themselves.

With regard to the original question of energy separation, discussed in chapter 2.4, energy separation due to the vortex shedding can be seen in the wake region of the flow in Figure 13.11. This is evidenced by the regions of high and low stagnation enthalpy. An important original observation is that for the first time energy separation can be seen clearly in the upper shear layer. Again this is demonstrated by regions of high and low stagnation enthalpy and is presumably associated with the Kelvin-Helmholtz instabilities. This is particularly important in that it implies that regions of vorticity exist along the shear layer, and also that energy is being transported from the base region across the shear layer into the main flow. Such a re-distribution of energy would undoubtedly modify base pressure values. The transport of energy across the shear layer may be due to the mechanism present in the vortex street, or a new mechanism particular to the Kelvin-Helmholtz instability due to the pressure differential across the shear layer.

Further investigation would need to be performed in order to quantify the magnitude of the energy separation and the effect on the pressure in the base region and the exact mechanism. Downstream of the base region, the energy separation can be seen to damp out considerably in a short distance, this can be attributed to the weak-coherence of the vortex street at the transonic speed.
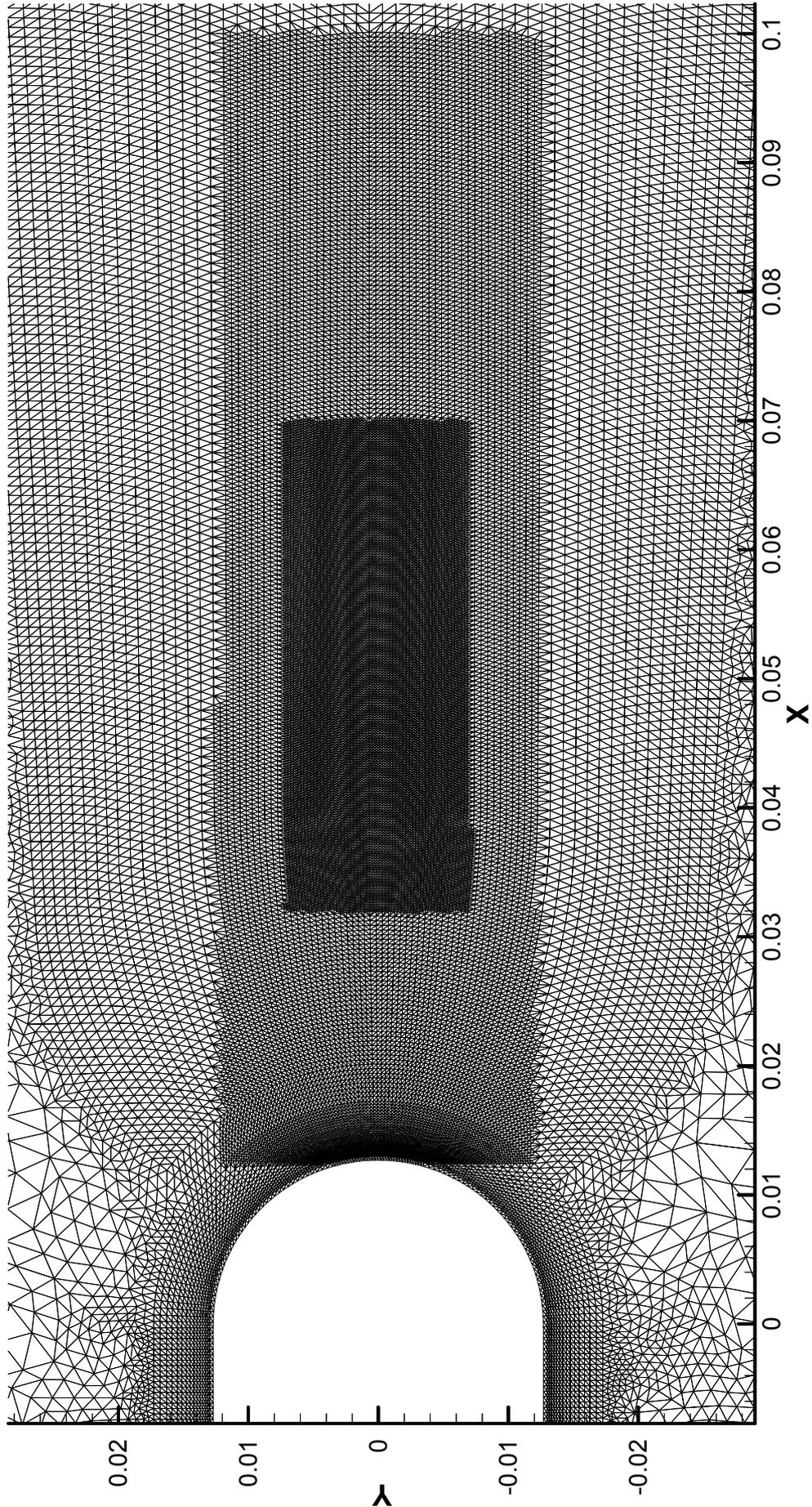
Figure 13.9: *Adapted Mesh for Time Resolved Flow, 3 Adaptive Steps*
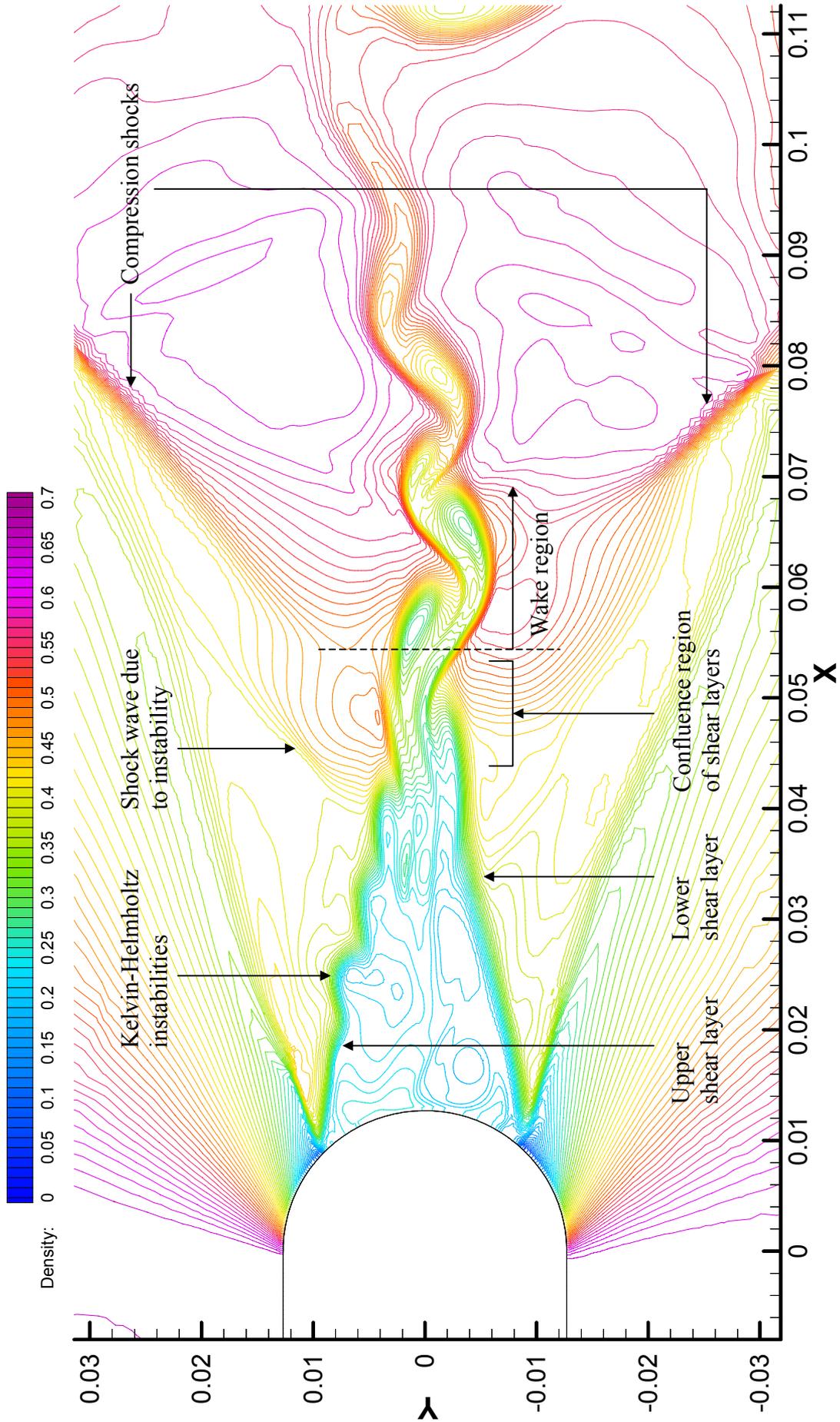
Figure 13.10: *Density Contours behind Trailing Edge of Flat Plate, $\Delta\rho = 0.01$.*
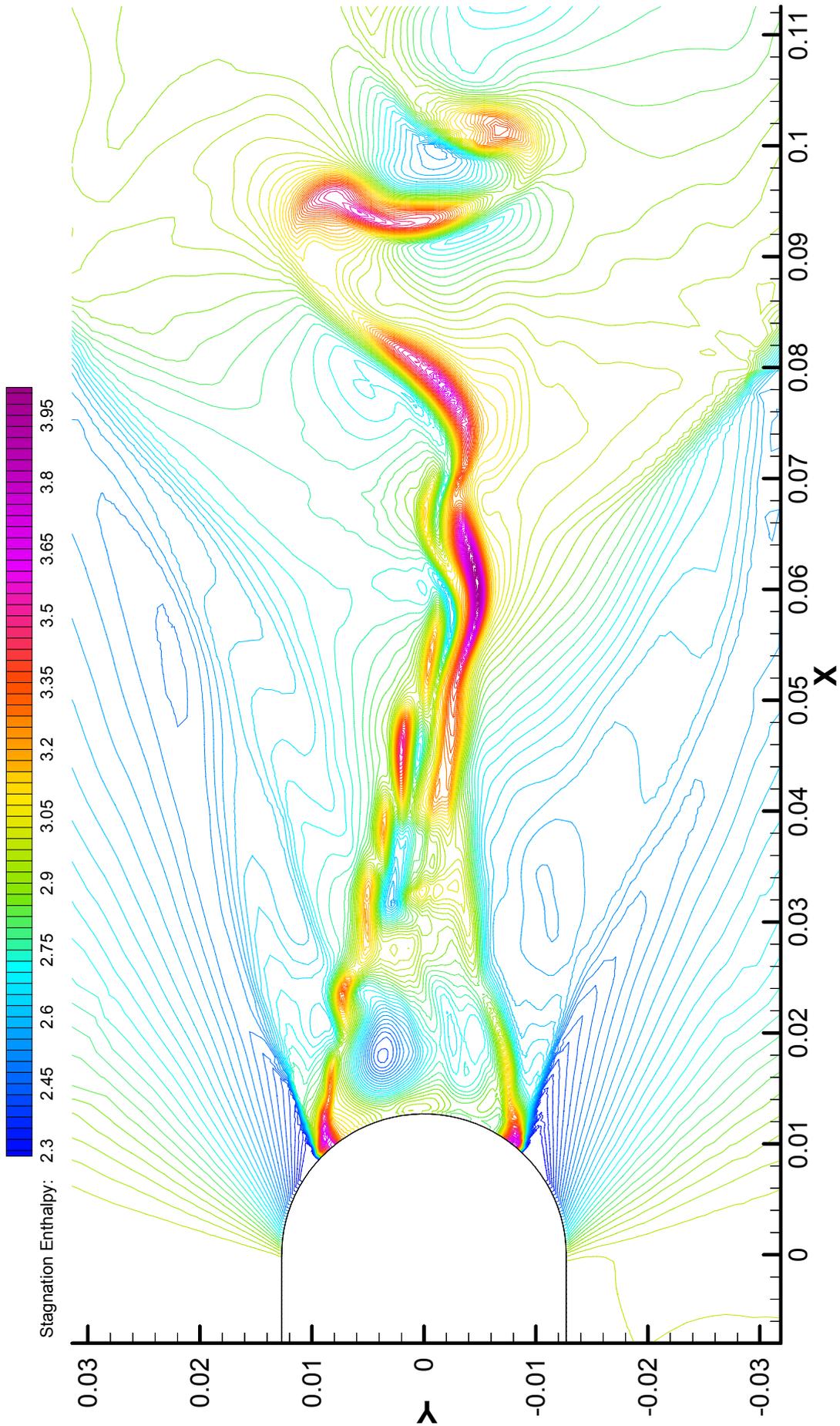
Figure 13.11: *Stagnation Enthalpy Contours behind Trailing Edge of Flat Plate, Δh = 0.025, Iteration 23000.*
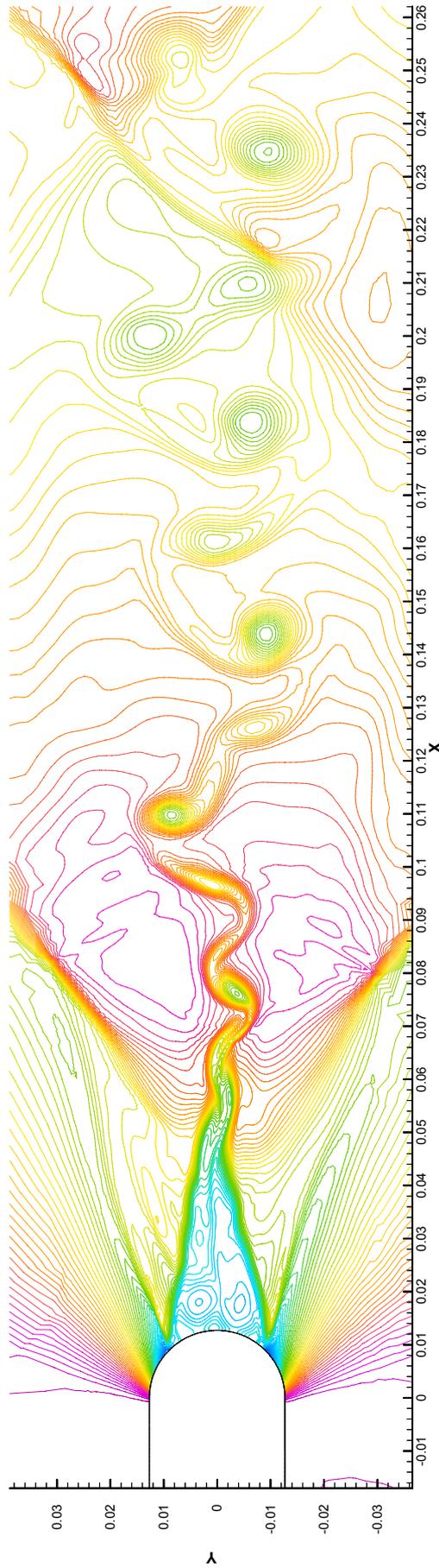
Figure 13.12: *Unsteady Flow Field behind Trailing Edge of Flat Plate, Iteration 22000, $\Delta\rho = 0.01$*
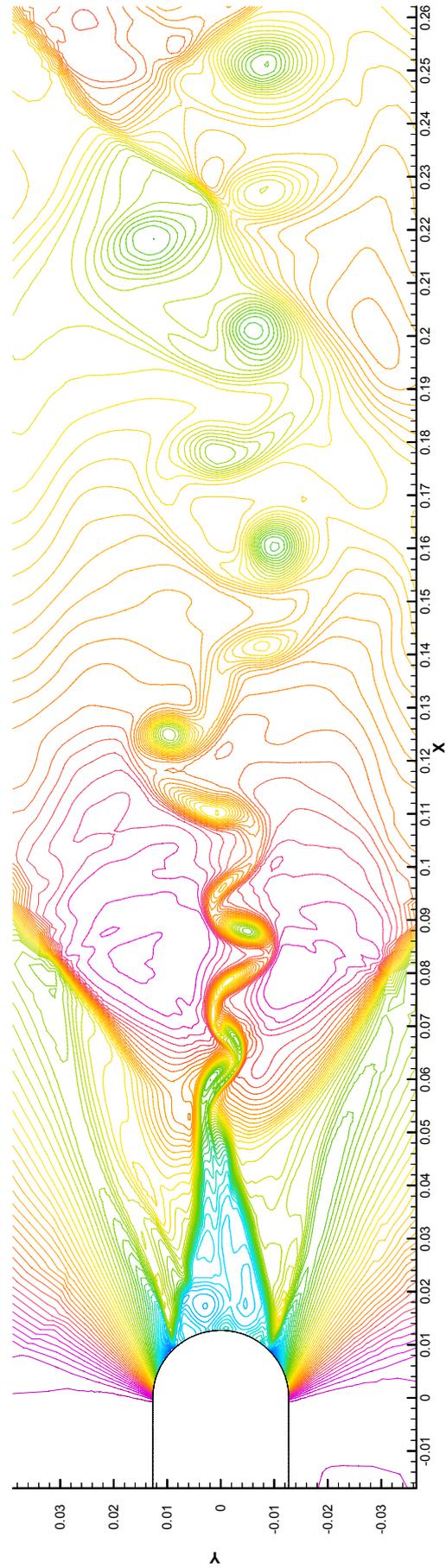
Figure 13.13: *Unsteady Flow Field behind Trailing Edge of Flat Plate, Iteration 22250, $\Delta\rho = 0.01$*
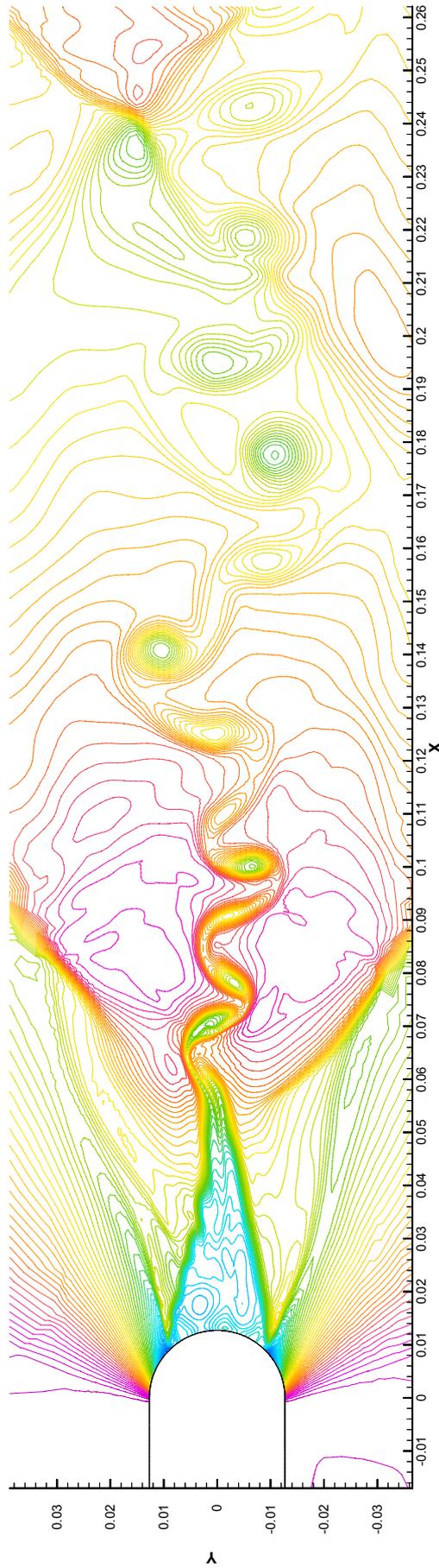
Figure 13.14: *Unsteady Flow Field behind Trailing Edge of Flat Plate, Iteration 22500, Δρ = 0.01*
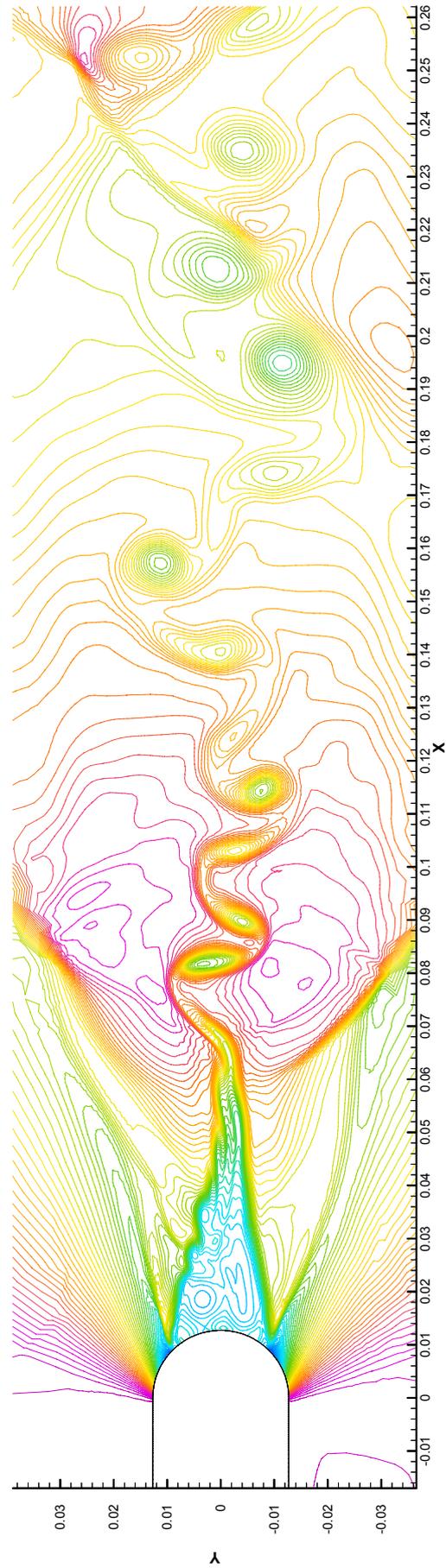
Figure 13.15: *Unsteady Flow Field behind Trailing Edge of Flat Plate, Iteration 22750, Δρ = 0.01*
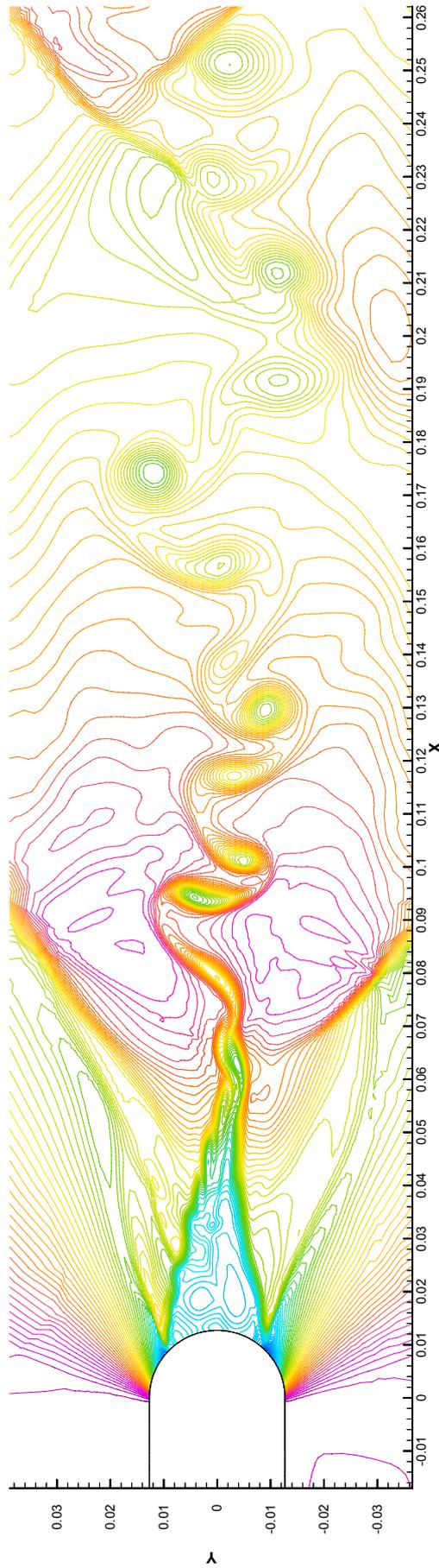
Figure 13.16: *Unsteady Flow Field behind Trailing Edge of Flat Plate, Iteration 23000, $\Delta\rho = 0.01$*
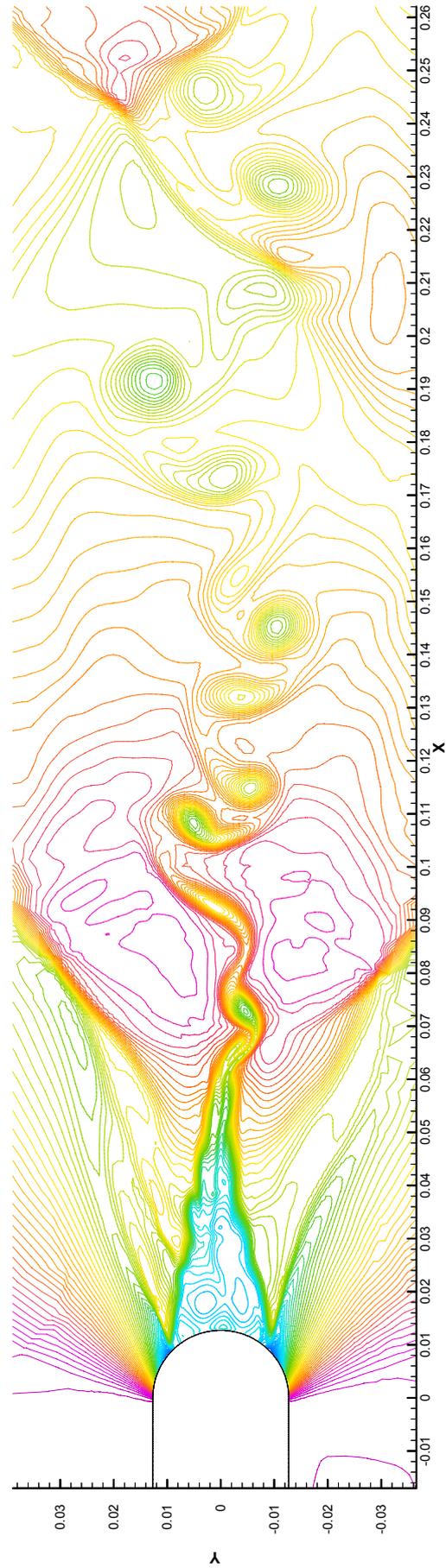
Figure 13.17: *Unsteady Flow Field behind Trailing Edge of Flat Plate, Iteration 23250, $\Delta\rho = 0.01$*

13.3   Preliminary Experimental Results

Preliminary experiments have been performed on a prototype model in the Hydrodynamics Laboratory. A photograph of the Water Table Model can be seen in Figure 13.18.

Figure 13.18: *Prototype Water Table Model*

The model consists of a nine-layer marine plywood base, sanded and varnished. The flat plate is machined from a square section of aluminium with a machined semi-circle at each end. The upper and lower walls are fashioned from angled aluminium. Where the sections of the wall are curved, the lower arm has been removed and the natural flexibility of the aluminium has been used to follow the desired line of the wall. This is only a prototype and tolerances for the outer walls are in the order of ±0.5mm; this is deemed acceptable by the author for the current study. Several instances of the unsteady flow field are given in Figure 13.19. The 'shock waves' stemming from behind the trailing edge can clearly be seen in all images[21]. In all images, Kelvin-Helmholtz instabilities are clearly visible along the shear layer projecting from the upper surface (pressure surface), and in some cases the instability is also present in the lower shear layer. This correlates well with the numerical data given in section 13.2. The

---

[21] Imaging: The images in Figure 13.19(a-f) were taken with a Kodak DC210+ Digital Camera. No special techniques were employed, the camera was simply held in the slightly off-vertical position above the trailing edge region of the flow.

fact that the instabilities are present in the experimental work strengthens the argument that the effects of surface tension may be ignored in the current study.

(a)   **(b)**

(c)   **(d)**

(e)   **(f)**
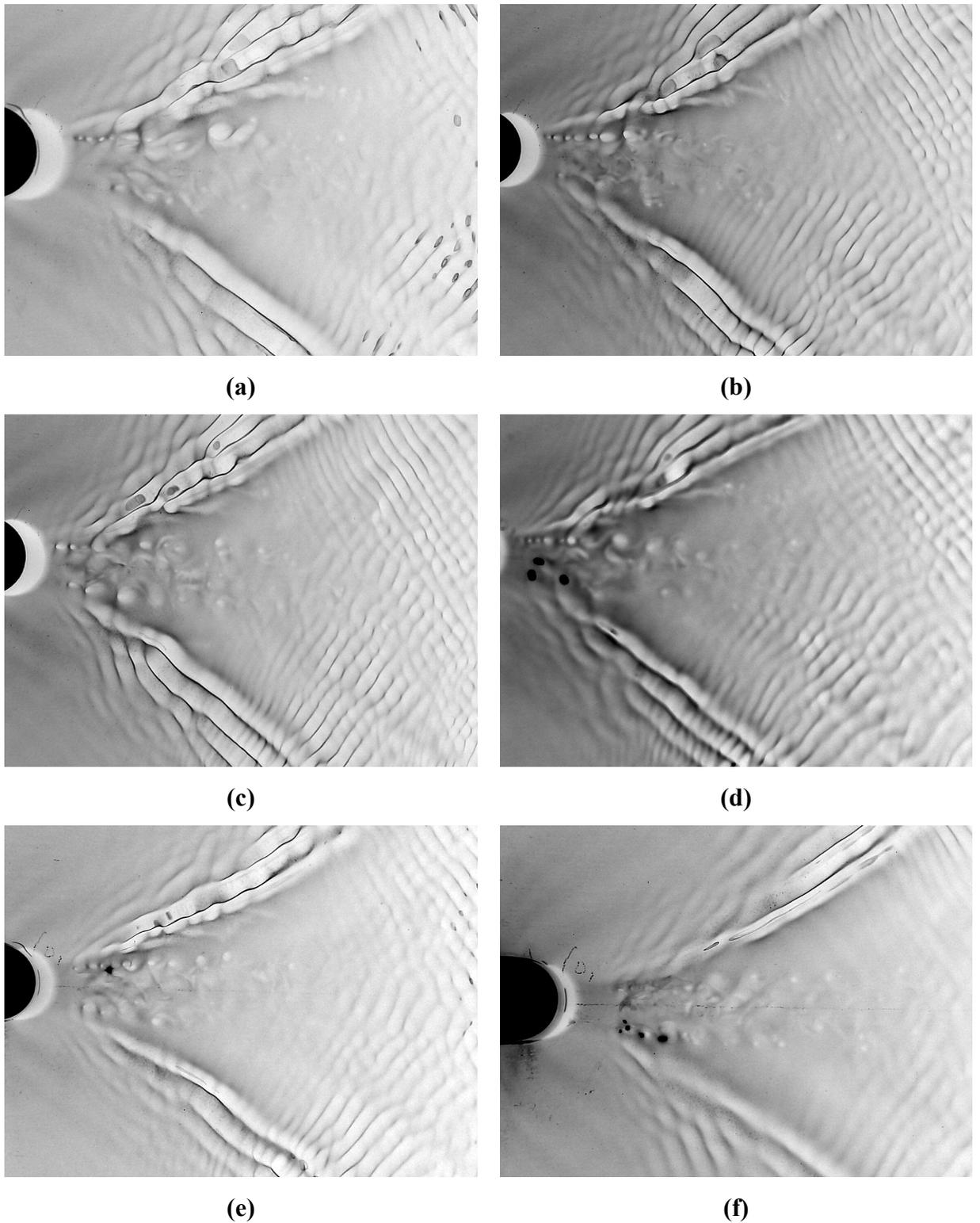
Figure 13.19: *Unsteady Flow Field Behind Flat Plate*

Previous experimental results for the equivalent test case can be seen in the Schlieren images in Figure 13.20(a)-(d); Carscallen & Gostelow (1994). A variety of the transient shedding patterns can clearly be seen and the definition of the shock structure in the trailing edge region is good. However, the structure of the triangular base region is unclear; the vortex shedding appears to emanate from the confluence point, and there is insufficient resolution along the shear layers to determine the existence of instabilities. The boundary layer on the suction surface is more prominent than that on the pressure surface.

(a)   (b)

(c)   (d)
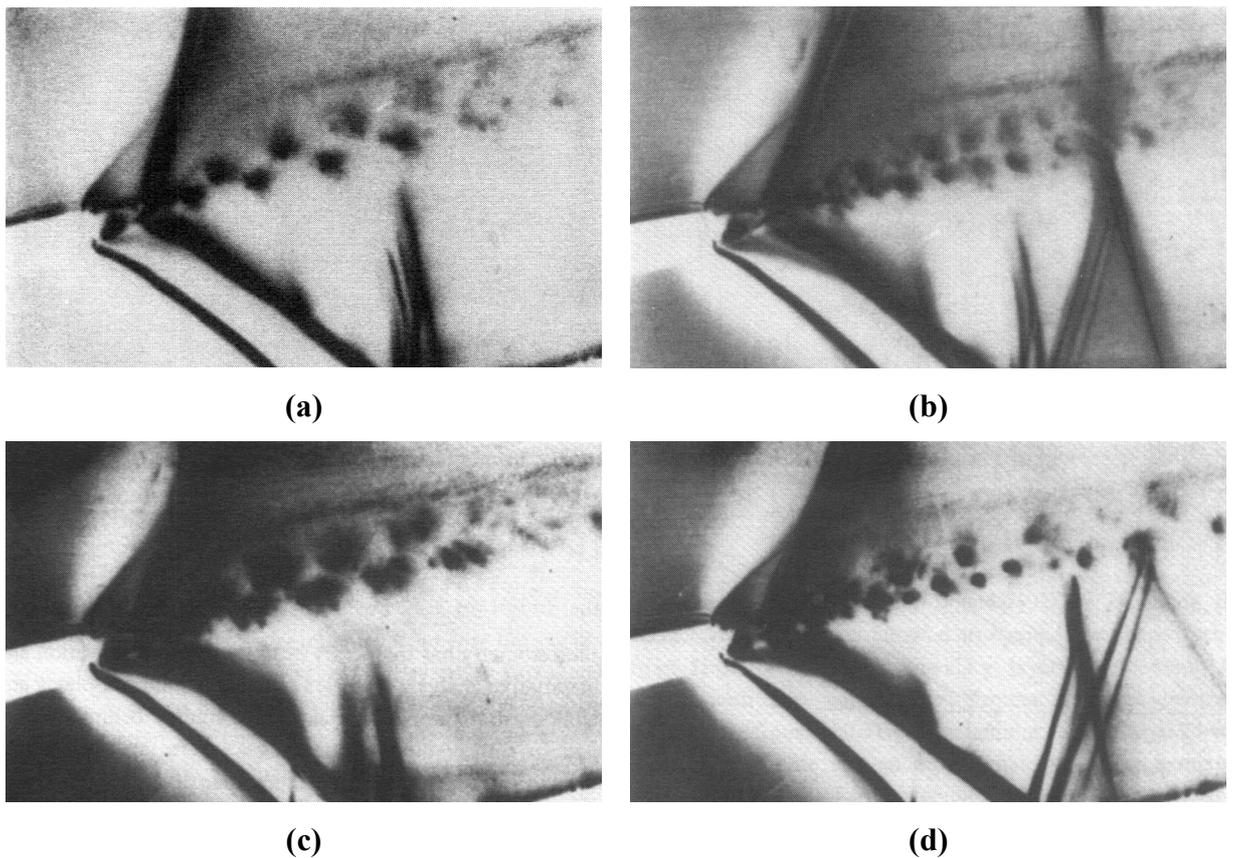
Figure 13.20: *Vortex Shedding Behind the Turbine Blade, Carscallen & Gostelow (1994)*

13.4   Concluding Remarks

A model has been presented for the water table geometry. Numerical results are in good agreement with the computed turbine blade Mach number distributions. The problems associated with the extension to the Navier-Stokes equations have been discussed; namely, the Reynolds number mismatch and the effects of surface tension. Unsteady computational results have clearly shown new unstable phenomena along the shear layers behind the trailing edge and their effects on the vortex shedding patterns. Two hypotheses have been presented for the mechanism behind the degeneration of the von Karman vortex street into transient

shedding patterns. The introduction of the instabilities adds a measure of vorticity along the shear layer. In order for the shear layer to remain linear, a sufficient level of counter-rotating vorticity must be introduced into the base pressure region to ensure that the total vorticity of the region is zero; if this is not the case, precession of the vortices will occur. The instabilities will therefore alter the characteristics of the flow in the base region.

An additional source of energy separation has been identified in the trailing edge region. Energy separation has also been shown, for the first time, to occur in the shear layers prior to the flow forming the familiar von Karman vortex street. This would have important implications for base pressure calculations and energy separation in the wake region.

It has been shown that these problems can also be addressed by experimental means. The water table results can be compared to those in Figure 13.20. Both exhibit similar shock structures in the trailing edge region. However, the water table results seem to imply that the instabilities may stem from the shear layers and convect into the wake region, whereas the experimental results of Carscallen & Gostelow show distinct vortex shedding from the confluence region. The unsteady phenomena in the water table results also appear less well defined than the corresponding experimental results; this may be because they are of a lower intensity or because the imaging technique is insufficient. These are all issues that may be addressed in future work.

Preliminary experiments have been most useful and show that further work on the experiment model is likely to provide an understanding of the unusual vortex shedding configurations observed in Figure 13.20.

14   CONCLUSIONS AND FURTHER WORK

A highly accurate numerical solver has been presented for the calculation of two-dimensional flow around turbine blading.

This has been based on application of computational fluid dynamics using an unstructured grid methodology. The choice of the unstructured grid is justified by the complex outer boundaries displayed in the water table geometry. This requires at least one layer of structure at the boundaries for the method to generate acceptable results but this is still easier than generating a full body-fitted grid. The unstructured method does allow the use of adaptive techniques, and the current technique has produced very good results. This may be further investigated by employing a technique of multi-blocking to allow both unstructured and structured zones within the domain. Mesh generation was relatively easy; the mesh is initially defined as a set of points on the domain boundary connected by an edge set. A Delaunay triangulation was then generated from this point set.

An analysis of several schemes was performed to find the most suitable for applying to the unstructured method. The author's own upwinding scheme based on the minmod limiter, the 'Quirk-Minmod' scheme, was found to generate the most accurate results. This was due to the higher order scheme being applied independently to each of the eigenvalues. In subsonic cases this does result in lower rates of convergence when compared to the Barth and Van Leer schemes. It is clear that there are still problems associated with the current scheme; in particular, work still needs to be done in examining the adiabatic properties of the wall boundaries, nevertheless the current scheme does offer the best performance over several other types of boundary.

A new subsonic inlet boundary condition was presented, based on the work of Manna (1992), with improvements to ensure that the boundary does not suffer from numerical inaccuracies when the inlet flow angle is tangential to the boundary. This is particularly important if circular or elliptical outer boundaries are chosen for a domain.

A complete set of test cases has been presented and validated that cover the range of flow phenomena that may be present in transonic two-dimensional flow.

As yet no proper testing has been performed on the viscous solver, as this was not considered essential for the generation of the Mach number distributions around the blades. Viscous

computations would make minimal differences to the Mach number distributions around the blades and the extension to fully viscous flow was not deemed necessary at this time. It was also shown that the inviscid scheme accurately depicted the flow around the turbine blade, particularly the positioning of the impinging shock on the suction surface.

The turbine blade in the study was chosen because it is currently under study at the University of Leicester and the CNRC. A great deal of computational and experimental data is available for further analysis if this is necessary; it is the technique of generating the analogous model that is of importance in this study. In particular, pressure distributions and Schlieren images were readily available. As such, the blade was found to be an acceptable choice.

It was decided that the axial direction would be used on which to base the Mach number distribution, however, the chord vector could also have been chosen. This would not have affected any computation of the blade, but would have resulted in a different geometry for the water table model. The main features of the flow would still be captured but would simply be in different regions of the geometry. In particular, the impinging shock on the suction surface would be further upstream. In this manner, the actual position of the shock is less important than the magnitude and characteristics of the shock.

The co-ordinates for the upper and lower walls were generated manually by an iterative method by the author. This was very time-consuming and a rather *trial and error* method. A further extension to this would be to develop an automated inverse method for generating the water table geometry. The author feels that this may well be very complex, but would be an exciting challenge for future study, and would make the code more useful for blade design purposes.

Energy separation has clearly been predicted behind the flat plate of the water table model, and by analogy behind turbine blades. This new phenomenon of energy separation along the free shear layers has been presented, and attributed to the presence of Kelvin-Helmholtz instabilities. There are now two, possibly coupled, mechanisms active in the trailing edge region that would have a combined effect on the base pressure and the overall energy separation.

14.1   Further Work

Computational work has so far been restricted to inviscid flow. The next step would be to run fully viscous simulations with a relevant turbulence model in order to capture the full dynamics of the system. Given the different Reynolds numbers involved the transitional

behaviour of the boundary layers would also need to be modelled. This should result in correct predictions of the boundary layers on either side of the blade. In particular, the fact that the pressure surface boundary layer is only one quarter the thickness of that on the suction surface should be predicted, Currie & Carscallen (1998), Arnone & Pacciani (1997). The correct thickness of boundary layers would be accompanied by the correct density and velocity gradients normal to the flow direction, this would give a more physically correct condition for the Kelvin-Helmholtz instabilities in the upper and lower shear layers to form. Their strengths and concentration would then be more accurately reproduced than in the inviscid case, as would the vortex shedding. The supposition that the Kelvin-Helmholtz instabilities are a mechanism for energy separation would also provide areas of further research for situations in which these instabilities are known to exist; in particular whether the mechanism is similar to that of vortex shedding or a new mechanism that needs to be explored.

With regard to the experimental aspects, Chapter 13.3 has clearly shown that the water table model is capable of showing the interaction between the vortex shedding and the shock waves. However, the model is very primitive, its purpose being merely to demonstrate the technique. Further experimental work would require a more complex model that would also allow more advanced visualisation methods. This would entail construction from transparent materials such as glass and Perspex and an improved approach to the flow visualisation. Ongoing work at the University of Leicester is orientated to producing more sophisticated models.

APPENDIX A: NEWTON-RAPHSON ALGORITHMS

The two routines listed below are those for solving for the Mach number at a subsonic inlet and a subsonic outlet.

Subsonic inlet

The subsonic inlet requires the defining of three flow variables at the boundary, namely: total pressure *Total_P*, total temperature *Total_T,* and inlet flow angle *inlet_angle*. The code to solve for Mach number using the Newton-Raphson algorithm is as follows:

```
void   Inlet_E_Sub( double* Q, complex vector, double* Q_out, matrix& E, matrix& DE )
{
   //     total_P                        upstream total pressure
   //     total_T                        upstream total temperature
   //     inlet_angle                    inlet flow angle in degrees

   double      angle = inlet_angle*PI/180;

   double      r0 = total_P / (total_T*UNIVERSAL_GAS_CONSTANT);
   double      a0 = sqrt( flow_gamma*UNIVERSAL_GAS_CONSTANT*total_T );
   double      r0grt = r0*a0;
   double      g1d2 = (flow_gamma+1)/2.0;
   double      refroe = r0;
   double      refa = a0;

   double      pr[MATRIX_SIZE];
   double      prb[MATRIX_SIZE];

   Change_QtoQ_P( Q, pr );                                        //   change to primitive

   complex     v_int( pr[1], pr[2] );
   double      U_int = dot( v_int, unit(vector) );
   double      V_int = -cross( v_int, unit(vector));
   double      c_int = sqrt( flow_gamma*pr[3]/pr[0] );
   double      m_int = U_int/c_int;
   double      a_int = pr[0]*c_int*sqr((m_int+1)/2.0);

   double      m_bnd = len(v_int)/c_int;
   double      err = 10E6;
   double      x_mb1, x_alt;
   double      f, fd;

   complex     flow_vect( cos(angle), sin(angle) );
   double      cos_flow = dot(vector, flow_vect);

   for (int iter=0; ((iter<20) AND (err > 10E-12)); iter++)
```

```
{
    x_alt = m_bnd;
    x_mb1 = 1.0+((flow_gamma-1)*sqr(m_bnd)/2.0);

    f = (a_int*pow(x_mb1, g1d2/(flow_gamma-1)))
 - (r0grt*sqr((cos_flow*m_bnd)+1.0)/4.0);

    fd = (a_int*m_bnd*g1d2*pow(x_mb1, ((3-flow_gamma)/(2.0*(flow_gamma-1)))))
 - (r0grt*cos_flow*((cos_flow*m_bnd)+1)/2.0);

    m_bnd -= (f/fd);
    err = fabs((m_bnd-x_alt)/m_bnd);
}
```

$$f = \left[\rho a(1+M_n)^2\right]_I \left(1+\frac{(\gamma-1)M^2}{2}\right)^{\frac{\gamma+1}{2(\gamma-1)}} - \left[p_0\sqrt{\frac{\gamma}{RT_0}}\right]_B (1+M\cos(\beta))^2$$

$$\partial f = \left[\rho a(1+M_n)^2\right]_I M \frac{\gamma+1}{2}\left(1+\frac{(\gamma-1)M^2}{2}\right)^{\frac{3-\gamma}{2(\gamma-1)}} - \left[p_0\sqrt{\frac{\gamma}{RT_0}}\right]_B \cos(\beta)(1+M\cos(\beta))$$

```
prb[0] = r0 / pow( (1.0+((flow_gamma-1)*sqr(m_bnd)/2.0)), (1.0/(flow_gamma-1)) );
prb[3] = total_P*pow( (prb[0]/r0), flow_gamma );


double      c_bnd = sqrt( flow_gamma*prb[3]/prb[0] );
double      uabs = m_bnd*c_bnd;
double      un = uabs*cos_flow;
double      u = uabs*cos(angle);
double      v = uabs*sin(angle);

prb[1] = u;
prb[2] = v;

Change_Q_PtoQ( prb, Q_out );                  //  return back to Q

//  Build E and DE from here for implicit method

double      E0 = prb[0]*un;
double      E1 = (prb[0]*u*un) + (prb[3]*real(vector));
double      E2 = (prb[0]*v*un) + (prb[3]*imag(vector));
double      E3 = un*((flow_gamma*prb[3]/(flow_gamma-1)) + (prb[0]*sqr(uabs)/2.0));

E.set( 0, 0, E0 );
E.set( 0, 1, E1 );
E.set( 0, 2, E2 );
E.set( 0, 3, E3 );

matrix     DH_P(1, MATRIX_SIZE);
matrix     DP_Q(MATRIX_SIZE, 1);

double      em = 1.0+((flow_gamma-1)*sqr(m_bnd)/2.0);
double      ema = pow(em, -(flow_gamma+1)/(2.0*(flow_gamma-1)) );
double      emb = pow(em, -flow_gamma/(flow_gamma-1) );
double      emd = pow(em, (1-(2.0*flow_gamma))/(flow_gamma-1) );

double      c3 = (1-sqr(m_bnd))/em;
double      c4 = -total_P*emd*flow_gamma*m_bnd;
double      c5 = cos_flow*(sqr(m_bnd)-2);
double      dE0 = (E0/m_bnd) * c3;

DH_P.set( 0, 0, dE0 );
DH_P.set( 0, 1, c4*( (cos(angle)*c5) + real(vector) ) );
DH_P.set( 0, 2, c4*( (sin(angle)*c5) + imag(vector) ) );
DH_P.set( 0, 3, dE0*flow_gamma*UNIVERSAL_GAS_CONSTANT*total_T/(flow_gamma-1) );

double      e0 = Q_out[3]/Q_out[0];                       //  total_energy
double      neta = flow_gamma*(flow_gamma-1)/(prb[0]*cube(c_bnd));

DP_Q.set( 0, 0, -neta*e0*uabs/2.0 );
```

$$\xi_m = 1+\frac{(\gamma-1)M^2}{2}$$

$$\psi_0 = \frac{\rho U}{M}\left(\frac{1-M^2}{\xi_m}\right)$$

$$\psi_1 = -p_0\xi_m^{\left(\frac{1-2\gamma}{\gamma-1}\right)}\gamma M\left(\left(\cos(\alpha)\cos(\beta)(M^2-2)\right)+\hat{n}_x\right)$$

$$\psi_2 = -p_0\xi_m^{\left(\frac{1-2\gamma}{\gamma-1}\right)}\gamma M\left(\left(\sin(\alpha)\cos(\beta)(M^2-2)\right)+\hat{n}_y\right)$$

$$\psi_3 = \psi_0\frac{\gamma RT_0}{(\gamma-1)}$$

$$\frac{\partial \boldsymbol{E}(\boldsymbol{P})}{\partial \boldsymbol{P}_B} = \begin{Bmatrix} 0 & \psi_0 & 0 & 0 \\ 0 & \psi_1 & 0 & 0 \\ 0 & \psi_2 & 0 & 0 \\ 0 & \psi_3 & 0 & 0 \end{Bmatrix}$$

```
        DP_Q.set( 1, 0, neta*e0*u/uabs  );
        DP_Q.set( 2, 0, neta*e0*v/uabs  );
        DP_Q.set( 3, 0, -neta*uabs/2.0  );

        DE = DH_P*DP_Q;
}
```

$$\frac{\partial \boldsymbol{P}_B}{\partial \boldsymbol{Q}} = \begin{cases} 0 & 0 & 0 & 0 \\ \phi_0 & \phi_1 & \phi_2 & \phi_3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{cases}$$

It is only necessary to store the non-zero row/column of these matrices as described in the chapter on boundary conditions.

C++ Code Segment: *Procedure to solve for Inlet Mach Number*

Q is an array of four doubles defining the inlet conservative variables, *vector* is the inlet normal vector, *Q_out* is the new vector of conservative variables, and *E* and *DE* are the flux and derivative of the flux defined as matrices. The matrix structure is simply a two-dimensional array of double.

Subsonic outlet

The outlet routine is very similar to the above routine returning the same variables. The outlet only requires the definition of the outlet static pressure *Static_P*.

```
void     Outlet_E_Sub( double* Q, complex vector, matrix& E, matrix& DE, double* Q_out )
{
    double          pr[MATRIX_SIZE];

    Change_QtoQ_P( Q, pr );

    //      static_P                        outlet static pressure

    complex      v_l( pr[1], pr[2] );
    double       U = dot(v_l, vector);
    double       V = -cross(v_l, vector);
    double       c_int = sqrt( flow_gamma*pr[3]/pr[0] );
    double       m_int = U/c_int;

    double       a_int = pr[0]*c_int*sqr(m_int+1);
    double       t_int = ((flow_gamma-1)*U) + (2.0*c_int);

    double       c3 = flow_gamma-1;
    double       c2 = 2.0*flow_gamma;
    double       c1 = flow_gamma+3.0;
    double       c0 = 2.0-(a_int*t_int/(flow_gamma*static_P));

    double       err = 10E6;
    double       xm, xm2, xm3, f200, df200;
    double       m_bnd = m_int;

    for (int iter=0; ((iter<20) AND (err > 10E-12)); iter++)
    {
        xm = m_bnd;
        xm2 = sqr(xm);
        xm3 = cube(xm);
        f = (c3*xm3)+(c2*xm2)+(c1*xm)+c0;
```

$$\alpha_3 = \gamma - 1$$
$$\alpha_2 = 2\gamma$$
$$\alpha_1 = \gamma + 3$$
$$\alpha_0 = 2 - \frac{\left[\rho a (1 + M_n)^2 \left((\gamma - 1)U + 2a\right)\right]_I}{\gamma p_\infty}$$

```
        df = (3.0*c3*xm2)+(2.0*c2*xm)+c1;
        m_bnd -= (f/df);
        err = fabs((m_bnd-xm)/m_bnd);
    }
```

$$f = \alpha_3 M_n^3 + \alpha_2 M_n^2 + \alpha_1 M_n + \alpha_0$$

$$\partial f = 3\alpha_3 M_n^2 + 2\alpha_2 M_n + M_n$$

```
    double      c_bnd = flow_gamma*static_P*sqr(m_bnd+1)/a_int;
    double      U_bnd = m_bnd*c_bnd;

    pr[0] = flow_gamma*static_P/(sqr(c_bnd));
    pr[1] = (real(vector)*U_bnd) - (imag(vector)*V);
    pr[2] = (imag(vector)*U_bnd) + (real(vector)*V);
    pr[3] = static_P;

    Change_Q_PtoQ( pr, Q_out );

    double      u = pr[1];
    double      v = pr[2];
    double      U_bar = sqr(u)+sqr(v);
    double      h = (flow_gamma*static_P/((flow_gamma-1)*pr[0]))+(U_bar/2.0);

    E.set( 0, 0, pr[0]*U_bnd  );
    E.set( 0, 1, (pr[0]*u*U_bnd)+(static_P*real(vector))  );
    E.set( 0, 2, (pr[0]*v*U_bnd)+(static_P*imag(vector))  );
    E.set( 0, 3, pr[0]*U_bnd*h  );

    DE.set( 0, 0, 0 );
    DE.set( 1, 0, real(vector) );
    DE.set( 2, 0, imag(vector) );
    DE.set( 3, 0, 0 );

    DE.set( 0, 1, -u*U_bnd );
    DE.set( 1, 1, U_bnd+(u*real(vector)) );
    DE.set( 2, 1, u*imag(vector) );
    DE.set( 3, 1, 0 );

    DE.set( 0, 2, -v*U_bnd );
    DE.set( 1, 2, v*real(vector) );
    DE.set( 2, 2, U_bnd+(v*imag(vector)) );
    DE.set( 3, 2, 0 );

    DE.set( 0, 3, -U_bnd*(h+(U_bar/2.0)) );
    DE.set( 1, 3, (h*real(vector))+(u*U_bnd) );
    DE.set( 2, 3, (h*imag(vector))+(v*U_bnd) );
    DE.set( 3, 3, 0 );
}
```

$$\frac{\partial E(\boldsymbol{Q})}{\partial \boldsymbol{Q}} = \left\{ \begin{array}{cccc} 0 & \hat{n}_x & \hat{n}_y & 0 \\ -uU_n & U_n + u\hat{n}_x & u\hat{n}_y & 0 \\ -vU_n & v\hat{n}_x & U_n + v\hat{n}_y & 0 \\ -U_n\left(h_0 + \dfrac{U^2}{2}\right) & h_0\hat{n}_x + uU_n & h_0\hat{n}_y + vU_n & 0 \end{array} \right\}$$

C++ Code Segment: *Procedure to solve for Outlet Mach Number*

APPENDIX B: CO-ORDINATE TABLES

Turbine Blade				Water Table Model				
Suction Surface		Pressure Surface		Pressure Surface		Suction Surface		
X	Y	X	Y	X	Y	X	Y	
0 -0.045927	0.048586	0 -0.045927	0.048586	0 -1.333333	0.4	0 -1.333333	-0.4	
1 -0.045865	0.049366	1 -0.045915	0.048322	1 -1.317169	0.4	1 -1.323333	-0.4	
2 -0.045747	0.050148	2 -0.045865	0.047528	2 -1.307169	0.4	2 -1.313333	-0.4	
3 -0.045581	0.05092	3 -0.045777	0.046741	3 -1.297169	0.4	3 -1.303333	-0.4	
4 -0.045348	0.051673	4 -0.045643	0.045962	4 -1.287169	0.4	4 -1.293333	-0.4	
5 -0.045055	0.052407	5 -0.045448	0.0452	5 -1.277169	0.4	5 -1.283333	-0.4	
6 -0.044886	0.052765	6 -0.045172	0.044461	6 -1.267169	0.4	6 -1.273333	-0.4	
7 -0.044717	0.053123	7 -0.044852	0.043737	7 -1.257169	0.4	7 -1.263333	-0.4	
8 -0.044528	0.053469	8 -0.044488	0.043036	8 -1.247169	0.4	8 -1.253333	-0.4	
9 -0.044339	0.053815	9 -0.044069	0.042369	9 -1.237169	0.4	9 -1.243333	-0.4	
10 -0.044116	0.054139	10 -0.04359	0.041741	10 -1.227169	0.4	10 -1.233333	-0.4	
11 -0.043892	0.054463	11 -0.043072	0.041143	11 -1.217169	0.4	11 -1.223333	-0.4	
12 -0.043643	0.05477	12 -0.042517	0.040583	12 -1.207169	0.4	12 -1.213333	-0.4	
13 -0.043394	0.055077	13 -0.041909	0.04008	13 -1.197169	0.4	13 -1.203333	-0.4	
14 -0.043127	0.05537	14 -0.041271	0.039613	14 -1.187169	0.4	14 -1.193333	-0.4	
15 -0.042861	0.055663	15 -0.040599	0.039201	15 -1.177169	0.4	15 -1.183333	-0.4	
16 -0.042576	0.055936	16 -0.039891	0.038849	16 -1.167169	0.4	16 -1.173333	-0.4	
17 -0.042291	0.056209	17 -0.039308	0.038585	17 -1.157169	0.4	17 -1.163333	-0.4	
18 -0.041677	0.056709	18 -0.038646	0.038345	18 -1.147169	0.4	18 -1.153333	-0.4	
19 -0.041044	0.057191	19 -0.037898	0.038055	19 -1.137169	0.4	19 -1.143333	-0.4	
20 -0.040394	0.057649	20 -0.037155	0.037757	20 -1.127169	0.4	20 -1.133333	-0.4	
21 -0.039729	0.058085	21 -0.036413	0.037452	21 -1.117169	0.4	21 -1.123333	-0.4	
22 -0.03905	0.058499	22 -0.035677	0.037134	22 -1.107169	0.4	22 -1.113333	-0.4	
23 -0.038355	0.058886	23 -0.034944	0.036809	23 -1.097169	0.4	23 -1.103333	-0.4	
24 -0.03765	0.059252	24 -0.034215	0.036475	24 -1.087169	0.4	24 -1.093333	-0.40001	
25 -0.036927	0.059581	25 -0.033492	0.036124	25 -1.077169	0.4	25 -1.083333	-0.400031	
26 -0.036192	0.059889	26 -0.032782	0.035748	26 -1.067169	0.4	26 -1.073333	-0.400045	
27 -0.035453	0.060184	27 -0.03208	0.035359	27 -1.057169	0.4	27 -1.063333	-0.400045	
28 -0.035077	0.060315	28 -0.031385	0.034958	28 -1.047169	0.4	28 -1.053333	-0.40002	
29 -0.034701	0.060446	29 -0.030698	0.034544	29 -1.037169	0.4	29 -1.043334	-0.399927	
30 -0.034323	0.06057	30 -0.030015	0.034122	30 -1.027169	0.4	30 -1.033335	-0.399807	
31 -0.033945	0.060695	31 -0.029343	0.033682	31 -1.017169	0.4	31 -1.023336	-0.39967	
32 -0.033564	0.060806	32 -0.028684	0.033222	32 -1.007169	0.4	32 -1.013337	-0.399507	
33 -0.033183	0.060918	33 -0.028037	0.032745	33 -0.997169	0.4	33 -1.003341	-0.399236	
34 -0.032796	0.061009	34 -0.027405	0.032252	34 -0.987169	0.4	34 -0.993815	-0.396571	
35 -0.03241	0.061099	35 -0.026776	0.031753	35 -0.977169	0.4	35 -0.984913	-0.392021	
36 -0.032019	0.061177	36 -0.02616	0.03124	36 -0.967169	0.4	36 -0.976292	-0.386957	
37 -0.031629	0.061254	37 -0.025549	0.030721	37 -0.957169	0.4	37 -0.967827	-0.381633	
38 -0.031237	0.061323	38 -0.024945	0.03019	38 -0.947169	0.4	38 -0.959499	-0.376098	
39 -0.030844	0.061391	39 -0.024364	0.029635	39 -0.937169	0.4	39 -0.951242	-0.370456	
40 -0.03045	0.061451	40 -0.023793	0.029071	40 -0.927169	0.4	40 -0.944596	-0.363062	

41	-0.030056	0.06151	41	-0.023236	0.028494	41	-0.917169	0.4	41	-0.938515	-0.355124
42	-0.029662	0.061557	42	-0.022683	0.027914	42	-0.907169	0.4	42	-0.932539	-0.347106
43	-0.029268	0.061604	43	-0.022143	0.027321	43	-0.897169	0.4	43	-0.926583	-0.339073
44	-0.028872	0.061631	44	-0.021604	0.026728	44	-0.887169	0.4	44	-0.92062	-0.331046
45	-0.028475	0.061658	45	-0.021078	0.026123	45	-0.877169	0.4	45	-0.914623	-0.323043
46	-0.028077	0.061671	46	-0.020551	0.025517	46	-0.867169	0.4	46	-0.908597	-0.315063
47	-0.027679	0.061685	47	-0.020052	0.024887	47	-0.857169	0.4	47	-0.902503	-0.307134
48	-0.027281	0.061691	48	-0.019558	0.024256	48	-0.847169	0.4	48	-0.896207	-0.299366
49	-0.026883	0.061697	49	-0.019075	0.023617	49	-0.837169	0.4	49	-0.889799	-0.291689
50	-0.026485	0.06169	50	-0.018592	0.022977	50	-0.827169	0.4	50	-0.883344	-0.284051
51	-0.026087	0.061683	51	-0.018123	0.022327	51	-0.817169	0.4	51	-0.876844	-0.276452
52	-0.025689	0.061666	52	-0.017656	0.021678	52	-0.807169	0.4	52	-0.870236	-0.268946
53	-0.025292	0.061649	53	-0.017192	0.021024	53	-0.797169	0.4	53	-0.86358	-0.261484
54	-0.024896	0.061616	54	-0.016735	0.020367	54	-0.787169	0.4	54	-0.856884	-0.254056
55	-0.0245	0.061583	55	-0.016279	0.019707	55	-0.777169	0.4	55	-0.85015	-0.246664
56	-0.024105	0.061535	56	-0.015843	0.019033	56	-0.767169	0.4	56	-0.842325	-0.240475
57	-0.02371	0.061487	57	-0.015409	0.01836	57	-0.757169	0.4	57	-0.833771	-0.235297
58	-0.023316	0.061429	58	-0.014985	0.017681	58	-0.747169	0.4	58	-0.825132	-0.23026
59	-0.022923	0.061371	59	-0.014561	0.017001	59	-0.737169	0.4	59	-0.816468	-0.225266
60	-0.022531	0.061301	60	-0.014143	0.016318	60	-0.727169	0.4	60	-0.807783	-0.220309
61	-0.022139	0.06123	61	-0.01373	0.015633	61	-0.717169	0.4	61	-0.799095	-0.215358
62	-0.021748	0.061152	62	-0.013317	0.014947	62	-0.707169	0.4	62	-0.790409	-0.210402
63	-0.021358	0.061074	63	-0.012909	0.014258	63	-0.697169	0.4	63	-0.781722	-0.20545
64	-0.020971	0.060984	64	-0.012503	0.013568	64	-0.687169	0.4	64	-0.773031	-0.200503
65	-0.020584	0.060894	65	-0.012098	0.012877	65	-0.677169	0.4	65	-0.764336	-0.195563
66	-0.020203	0.060783	66	-0.011706	0.012177	66	-0.667169	0.4	66	-0.75563	-0.190644
67	-0.019822	0.060672	67	-0.011316	0.011478	67	-0.657169	0.4	67	-0.747627	-0.185667
68	-0.019443	0.060549	68	-0.010932	0.010776	68	-0.647169	0.400003	68	-0.740067	-0.179121
69	-0.019064	0.060427	69	-0.01055	0.010073	69	-0.637169	0.400012	69	-0.732547	-0.17253
70	-0.018688	0.060297	70	-0.010167	0.009369	70	-0.627169	0.400018	70	-0.725046	-0.165916
71	-0.018311	0.060166	71	-0.009792	0.008662	71	-0.617169	0.400026	71	-0.717555	-0.159292
72	-0.017938	0.060026	72	-0.009417	0.007955	72	-0.607169	0.400025	72	-0.710074	-0.152657
73	-0.017565	0.059887	73	-0.009043	0.007248	73	-0.597278	0.399224	73	-0.702622	-0.145988
74	-0.017196	0.059737	74	-0.008672	0.006538	74	-0.587892	0.395789	74	-0.694468	-0.140342
75	-0.016828	0.059587	75	-0.008303	0.005828	75	-0.578859	0.391506	75	-0.685539	-0.135843
76	-0.016467	0.059422	76	-0.007934	0.005118	76	-0.570095	0.386697	76	-0.676408	-0.131767
77	-0.016105	0.059257	77	-0.007573	0.004403	77	-0.561594	0.381433	77	-0.667163	-0.127955
78	-0.015749	0.05908	78	-0.007214	0.003687	78	-0.553287	0.375869	78	-0.657826	-0.124376
79	-0.015394	0.058902	79	-0.006856	0.002971	79	-0.545575	0.369545	79	-0.648428	-0.120958
80	-0.015043	0.058713	80	-0.006503	0.002253	80	-0.538134	0.362865	80	-0.63892	-0.117863
81	-0.014693	0.058524	81	-0.006151	0.001535	81	-0.53069	0.356187	81	-0.629317	-0.115074
82	-0.014346	0.058327	82	-0.005798	0.000816	82	-0.523244	0.349512	82	-0.619644	-0.112539
83	-0.014	0.05813	83	-0.005451	0.000095	83	-0.515777	0.342861	83	-0.60994	-0.110123
84	-0.013657	0.057926	84	-0.005104	-0.000626	84	-0.508318	0.3362	84	-0.600214	-0.1078
85	-0.013315	0.057723	85	-0.004756	-0.001347	85	-0.500822	0.329582	85	-0.590414	-0.105813
86	-0.01298	0.057509	86	-0.004412	-0.00207	86	-0.49268	0.323783	86	-0.580584	-0.103978
87	-0.012645	0.057295	87	-0.00407	-0.002793	87	-0.48422	0.318455	87	-0.570741	-0.102211
88	-0.012318	0.057069	88	-0.003727	-0.003516	88	-0.475663	0.313286	88	-0.560887	-0.100506
89	-0.011991	0.056843	89	-0.003388	-0.004241	89	-0.467017	0.308262	89	-0.551033	-0.098805
90	-0.011673	0.056605	90	-0.003052	-0.004968	90	-0.458237	0.303476	90	-0.541187	-0.097056
91	-0.011355	0.056366	91	-0.002717	-0.005695	91	-0.449459	0.298688	91	-0.531341	-0.095311

92	-0.011045	0.056115	92	-0.002384	-0.006422	92	-0.440475	0.294299	92	-0.526415	-0.094452
93	-0.010736	0.055865	93	-0.002052	-0.007151	93	-0.43142	0.290056	93	-0.521489	-0.093593
94	-0.010427	0.055612	94	-0.001722	-0.00788	94	-0.422309	0.285933	94	-0.51656	-0.09275
95	-0.010119	0.05536	95	-0.001393	-0.008609	95	-0.413208	0.281789	95	-0.511632	-0.091908
96	-0.009822	0.055094	96	-0.001067	-0.00934	96	-0.404113	0.277633	96	-0.506557	-0.091065
97	-0.009526	0.054828	97	-0.000742	-0.010071	97	-0.394996	0.273525	97	-0.501766	-0.090281
98	-0.00923	0.05456	98	-0.000418	-0.010803	98	-0.385858	0.269462	98	-0.497106	-0.089612
99	-0.008935	0.054293	99	-0.000096	-0.011536	99	-0.376732	0.265373	99	-0.491854	-0.088964
100	-0.00865	0.054017	100	0.000225	-0.012269	100	-0.36756	0.26139	100	-0.487639	-0.088574
101	-0.008364	0.05374	101	0.000545	-0.013002	101	-0.358382	0.257419	101	-0.48189	-0.088127
102	-0.00809	0.053453	102	0.000862	-0.013737	102	-0.349502	0.253693	102	-0.471902	-0.087639
103	-0.007816	0.053165	103	0.001178	-0.014472	103	-0.340667	0.249007	103	-0.461908	-0.087301
104	-0.007554	0.052866	104	0.001492	-0.015208	104	-0.331878	0.244242	104	-0.451911	-0.087042
105	-0.007292	0.052567	105	0.001805	-0.015945	105	-0.323156	0.239352	105	-0.441913	-0.086844
106	-0.007038	0.052261	106	0.002117	-0.016682	106	-0.314619	0.234145	106	-0.431915	-0.08666
107	-0.006783	0.051954	107	0.002428	-0.017419	107	-0.306201	0.228756	107	-0.421916	-0.086487
108	-0.006535	0.051642	108	0.002738	-0.018157	108	-0.298103	0.223014	108	-0.411918	-0.086333
109	-0.006287	0.05133	109	0.003045	-0.018896	109	-0.290377	0.216665	109	-0.401918	-0.086198
110	-0.006043	0.051015	110	0.003351	-0.019636	110	-0.282631	0.210341	110	-0.396919	-0.086142
111	-0.005799	0.0507	111	0.003655	-0.020376	111	-0.27489	0.20401	111	-0.391919	-0.086086
112	-0.005567	0.050376	112	0.003959	-0.021116	112	-0.267051	0.197801	112	-0.386919	-0.086047
113	-0.005335	0.050051	113	0.004261	-0.021857	113	-0.25924	0.191558	113	-0.384419	-0.086028
114	-0.005103	0.049727	114	0.004563	-0.022598	114	-0.251486	0.185244	114	-0.381919	-0.086009
115	-0.004871	0.049403	115	0.004863	-0.02334	115	-0.243411	0.179354	115	-0.379374	-0.086101
116	-0.00465	0.049071	116	0.00516	-0.024083	116	-0.235286	0.173525	116	-0.376828	-0.086194
117	-0.004429	0.048739	117	0.005456	-0.024827	117	-0.2272	0.167641	117	-0.374392	-0.08644
118	-0.00421	0.048406	118	0.005751	-0.025571	118	-0.219021	0.161895	118	-0.371955	-0.086686
119	-0.003992	0.048073	119	0.006044	-0.026315	119	-0.210718	0.156321	119	-0.36984	-0.087114
120	-0.003788	0.047732	120	0.006337	-0.02706	120	-0.202234	0.151028	120	-0.367725	-0.087542
121	-0.003584	0.047391	121	0.006628	-0.027805	121	-0.193619	0.145956	121	-0.364968	-0.088223
122	-0.003391	0.047043	122	0.006919	-0.028551	122	-0.184878	0.141103	122	-0.362211	-0.088903
123	-0.003197	0.046694	123	0.007207	-0.029297	123	-0.176029	0.136451	123	-0.359891	-0.089538
124	-0.003008	0.046343	124	0.007494	-0.030044	124	-0.167006	0.13214	124	-0.357571	-0.090172
125	-0.00282	0.045992	125	0.00778	-0.030792	125	-0.158019	0.127755	125	-0.355074	-0.090875
126	-0.002639	0.045636	126	0.008065	-0.03154	126	-0.149045	0.123349	126	-0.352576	-0.091578
127	-0.002458	0.04528	127	0.008348	-0.032288	127	-0.139852	0.119412	127	-0.350316	-0.092122
128	-0.00228	0.044923	128	0.008631	-0.033037	128	-0.134432	0.117053	128	-0.348055	-0.092665
129	-0.002102	0.044565	129	0.008912	-0.033786	129	-0.130691	0.115405	129	-0.346085	-0.09298
130	-0.001929	0.044206	130	0.009191	-0.034536	130	-0.124378	0.112576	130	-0.344114	-0.093295
131	-0.001755	0.043848	131	0.009468	-0.035287	131	-0.121562	0.111323	131	-0.342036	-0.093369
132	-0.001595	0.043483	132	0.009744	-0.036038	132	-0.117342	0.109631	132	-0.339959	-0.093442
133	-0.001435	0.043119	133	0.010019	-0.036789	133	-0.112275	0.107627	133	-0.33412	-0.093434
134	-0.00128	0.042751	134	0.010293	-0.037541	134	-0.106959	0.105637	134	-0.33023	-0.093332
135	-0.001124	0.042383	135	0.010565	-0.038294	135	-0.102862	0.104266	135	-0.324125	-0.09314
136	-0.000972	0.042014	136	0.010837	-0.039046	136	-0.097898	0.103006	136	-0.31413	-0.092819
137	-0.00082	0.041645	137	0.011108	-0.039799	137	-0.09312	0.102046	137	-0.304135	-0.092498
138	-0.000673	0.041274	138	0.011377	-0.040553	138	-0.087753	0.101241	138	-0.294141	-0.092153
139	-0.000526	0.040902	139	0.011644	-0.041307	139	-0.08322	0.100656	139	-0.284152	-0.091688
140	-0.000382	0.04053	140	0.01191	-0.042062	140	-0.080908	0.100444	140	-0.277717	-0.091341
141	-0.000237	0.040158	141	0.012175	-0.042817	141	-0.073266	0.099709	141	-0.274166	-0.091157
142	-0.000095	0.039784	142	0.012439	-0.043572	142	-0.067793	0.099219	142	-0.26417	-0.090928

143	0.000046	0.03941	143	0.012703	-0.044328	143	-0.063302	0.098881	143	-0.25417	-0.090817
144	0.000185	0.039037	144	0.012965	-0.045084	144	-0.058343	0.098574	144	-0.244171	-0.090715
145	0.000324	0.038663	145	0.013227	-0.04584	145	-0.053384	0.098267	145	-0.234171	-0.09062
146	0.000452	0.038286	146	0.013486	-0.046597	146	-0.053353	0.09825	146	-0.224172	-0.090531
147	0.00058	0.037908	147	0.013744	-0.047355	147	-0.053323	0.098233	147	-0.214172	-0.090446
148	0.000703	0.037528	148	0.014001	-0.048112	148	-0.050974	0.098151	148	-0.204172	-0.090366
149	0.000826	0.037148	149	0.014258	-0.04887	149	-0.048626	0.098068	149	-0.194173	-0.090292
150	0.000947	0.036767	150	0.014512	-0.049629	150	-0.045979	0.098034	150	-0.184173	-0.090221
151	0.001069	0.036386	151	0.014766	-0.050388	151	-0.043332	0.098	151	-0.174173	-0.090154
152	0.001188	0.036005	152	0.01502	-0.051146	152	-0.038332	0.098	152	-0.164173	-0.090088
153	0.001308	0.035623	153	0.015274	-0.051905	153	-0.033332	0.098	153	-0.154174	-0.090026
154	0.001426	0.035241	154	0.015526	-0.052665	154	-0.023332	0.098	154	-0.144174	-0.089983
155	0.001545	0.034859	155	0.015776	-0.053425	155	-0.013332	0.098	155	-0.134174	-0.089973
156	0.001664	0.034478	156	0.016027	-0.054184	156	-0.003332	0.098	156	-0.124174	-0.089975
157	0.001782	0.034096	157	0.016278	-0.054944	157	0.006668	0.098	157	-0.114174	-0.089985
158	0.001899	0.033714	158	0.016529	-0.055704	158	0.016668	0.098	158	-0.104174	-0.089997
159	0.002016	0.033331	159	0.016777	-0.056464	159	0.021668	0.098	159	-0.094174	-0.09
160	0.002133	0.032948	160	0.017026	-0.057225	160	0.026668	0.098	160	-0.084174	-0.09
161	0.00225	0.032566	161	0.017274	-0.057985	161	0.031668	0.098	161	-0.074174	-0.09
162	0.002367	0.032184	162	0.017523	-0.058746	162	0.036668	0.098	162	-0.064174	-0.09
163	0.002484	0.031801	163	0.017764	-0.05951	163	0.041668	0.098	163	-0.054174	-0.09
164	0.002597	0.031418	164	0.018005	-0.060273	164	0.046668	0.098	164	-0.044174	-0.09
165	0.002711	0.031035	165	0.018245	-0.061036	165	0.051668	0.097998	165	-0.034174	-0.09
166	0.002824	0.030651	166	0.018485	-0.061799	166	0.056668	0.097995	166	-0.024174	-0.09
167	0.002937	0.030268	167	0.018726	-0.062562	167	0.061668	0.097992	167	-0.014174	-0.09
168	0.003161	0.0295	168	0.018966	-0.063325	168	0.066668	0.097989	168	-0.004174	-0.09
169	0.003385	0.028732	169	0.019206	-0.064089	169	0.071668	0.097987	169	0.005826	-0.09
170	0.003608	0.027964	170	0.019442	-0.064854	170	0.076668	0.097984	170	0.015826	-0.09
171	0.00383	0.027196	171	0.019677	-0.065618	171	0.081668	0.09798	171	0.020826	-0.09
172	0.004053	0.026427	172	0.019912	-0.066383	172	0.086668	0.097977	172	0.025826	-0.09
173	0.004276	0.025659	173	0.020148	-0.067147	173	0.091668	0.097974	173	0.030826	-0.09
174	0.004498	0.024891	174	0.020383	-0.067912	174	0.096668	0.097971	174	0.035826	-0.09
175	0.004719	0.024122	175	0.020619	-0.068676	175	0.101667	0.09792	175	0.040826	-0.09
176	0.004939	0.023353	176	0.020854	-0.069441	176	0.106667	0.09787	176	0.045826	-0.09
177	0.005159	0.022584	177	0.02109	-0.070206	177	0.111667	0.09787	177	0.050826	-0.09
178	0.005379	0.021815	178	0.021325	-0.070971	178	0.116667	0.09787	178	0.055826	-0.09
179	0.005598	0.021045	179	0.021557	-0.071737	179	0.121667	0.09787	179	0.060826	-0.09
180	0.005817	0.020276	180	0.021788	-0.072503	180	0.126667	0.09787	180	0.065826	-0.09
181	0.006036	0.019506	181	0.022019	-0.073268	181	0.131667	0.09787	181	0.070826	-0.09
182	0.006145	0.019121	182	0.022251	-0.074034	182	0.136667	0.09787	182	0.075826	-0.09
183	0.006254	0.018737	183	0.022482	-0.0748	183	0.146667	0.09787	183	0.080826	-0.09
184	0.006363	0.018352	184	0.022713	-0.075566	184	0.156667	0.09787	184	0.085826	-0.09
185	0.006473	0.017967	185	0.022945	-0.076332	185	0.166667	0.09787	185	0.090826	-0.09
186	0.006582	0.017583	186	0.023176	-0.077097	186	0.176667	0.09787	186	0.095826	-0.09
187	0.006691	0.017198	187	0.023407	-0.077863	187	0.186667	0.09787	187	0.100826	-0.09
188	0.006799	0.016813	188	0.023639	-0.078629	188	0.196667	0.09787	188	0.105826	-0.09
189	0.006908	0.016428	189	0.023868	-0.079396	189	0.206667	0.09787	189	0.110826	-0.09
190	0.007017	0.016043	190	0.024097	-0.080163	190	0.216667	0.09787	190	0.115826	-0.09
191	0.007125	0.015658	191	0.024325	-0.080929	191	0.226667	0.09787	191	0.118326	-0.09
192	0.007233	0.015273	192	0.024554	-0.081696	192	0.236667	0.09787	192	0.120826	-0.09
193	0.007341	0.014888	193	0.024782	-0.082463	193	0.246667	0.09787	193	0.123326	-0.09

194	0.007449	0.014503	194	0.025011	-0.083229	194	0.256667	0.09787	194	0.125826	-0.09
195	0.007558	0.014118	195	0.025239	-0.083996	195	0.266667	0.09787	195	0.128326	-0.09
196	0.007666	0.013733	196	0.025468	-0.084763	196	0.276667	0.09787	196	0.130826	-0.09
197	0.007775	0.013348	197	0.025696	-0.085529	197	0.286667	0.09787	197	0.133326	-0.09
198	0.007883	0.012963	198	0.025925	-0.086296	198	0.296667	0.09787	198	0.135826	-0.09
199	0.007991	0.012577	199	0.026151	-0.087064	199	0.306667	0.09787	199	0.140826	-0.09
200	0.008099	0.012192	200	0.026376	-0.087832	200	0.316667	0.09787	200	0.145826	-0.09
201	0.008208	0.011807	201	0.026601	-0.088599	201	0.326667	0.09787	201	0.150826	-0.09
202	0.008317	0.011422	202	0.026826	-0.089367	202	0.336667	0.09787	202	0.155826	-0.09
203	0.008425	0.011037	203	0.027051	-0.090135	203	0.346667	0.09787	203	0.165826	-0.09
204	0.008533	0.010652	204	0.027274	-0.090903	204	0.356667	0.09787	204	0.175826	-0.09
205	0.008641	0.010267	205	0.027498	-0.091671	205	0.366667	0.09787	205	0.185826	-0.09
206	0.00875	0.009882	206	0.027721	-0.09244	206	0.376667	0.09787	206	0.195826	-0.09
207	0.008858	0.009497	207	0.027833	-0.092824	207	0.386667	0.09787	207	0.205826	-0.09
208	0.008965	0.009111	208	0.027945	-0.093208	208	0.396667	0.09787	208	0.215826	-0.09
209	0.009071	0.008726	209	0.028168	-0.093976	209	0.406667	0.09787	209	0.225826	-0.09
210	0.009178	0.008341	210	0.028389	-0.094745	210	0.416667	0.09787	210	0.235826	-0.09
211	0.009285	0.007956	211	0.028611	-0.095514	211	0.426667	0.09787	211	0.245826	-0.09
212	0.009392	0.00757	212	0.028722	-0.095898	212	0.436667	0.09787	212	0.255826	-0.09
213	0.009498	0.007185	213	0.028832	-0.096283	213	0.446667	0.09787	213	0.265826	-0.09
214	0.009605	0.006799	214	0.028942	-0.096667	214	0.456667	0.09787	214	0.275826	-0.09
215	0.009712	0.006414	215	0.029053	-0.097052	215	0.466667	0.09787	215	0.285826	-0.09
216	0.009819	0.006029	216	0.029274	-0.09782	216	0.476667	0.09787	216	0.295826	-0.09
217	0.009925	0.005643	217	0.029496	-0.098589	217	0.486667	0.09787	217	0.305826	-0.09
218	0.010032	0.005258	218	0.029717	-0.099358	218	0.496667	0.09787	218	0.315826	-0.09
219	0.010139	0.004872	219	0.029938	-0.100127	219	0.506667	0.09787	219	0.325826	-0.09
220	0.010352	0.004101	220	0.03016	-0.100896	220	0.516667	0.09787	220	0.335826	-0.09
221	0.010566	0.00333	221	0.03027	-0.10128	221	0.526667	0.09787	221	0.345826	-0.09
222	0.010779	0.002559	222	0.03038	-0.101665	222	0.536667	0.09787	222	0.355826	-0.09
223	0.010992	0.001788	223	0.030489	-0.10205	223	0.546667	0.09787	223	0.365826	-0.09
224	0.011204	0.001016	224	0.030598	-0.102435	224	0.556667	0.09787	224	0.375826	-0.09
225	0.01131	0.00063	225	0.030707	-0.10282	225	0.566667	0.09787	225	0.385826	-0.09
226	0.011415	0.000245	226	0.030817	-0.103204	226	0.576667	0.09787	226	0.395826	-0.09
227	0.011521	-0.000141	227	0.030927	-0.103589	227	0.586667	0.09787	227	0.405826	-0.09
228	0.011627	-0.000527	228	0.031036	-0.103974	228	0.596667	0.09787	228	0.415826	-0.09
229	0.011733	-0.000912	229	0.031145	-0.104358	229	0.606667	0.09787	229	0.425826	-0.09
230	0.011839	-0.001298	230	0.031255	-0.104743	230	0.616667	0.09787	230	0.435826	-0.09
231	0.011945	-0.001684	231	0.031364	-0.105128	231	0.626667	0.09787	231	0.445826	-0.09
232	0.01205	-0.00207	232	0.031473	-0.105513	232	0.636667	0.09787	232	0.455826	-0.09
233	0.012155	-0.002456	233	0.031582	-0.105898	233	0.646667	0.09787	233	0.465826	-0.09
234	0.012261	-0.002841	234	0.031692	-0.106282	234	0.656667	0.09787	234	0.475826	-0.09
235	0.012366	-0.003227	235	0.031801	-0.106667	235	0.666667	0.09787	235	0.485826	-0.09
236	0.012471	-0.003613	236	0.03191	-0.107052				236	0.495826	-0.09
237	0.012576	-0.003999	237	0.032019	-0.107437				237	0.505826	-0.09
238	0.012681	-0.004385	238	0.032127	-0.107823				238	0.515826	-0.09
239	0.012786	-0.004771	239	0.032233	-0.108207				239	0.525826	-0.09
240	0.012891	-0.005157	240	0.032339	-0.108592				240	0.535826	-0.09
241	0.012996	-0.005543	241	0.032456	-0.108974				241	0.545826	-0.09
242	0.0131	-0.005929	242	0.032573	-0.109355				242	0.555826	-0.09
243	0.013203	-0.006315	243	0.032693	-0.109735				243	0.565826	-0.09
244	0.013306	-0.006701	244	0.032813	-0.110115				244	0.575826	-0.09

245	0.013409	-0.007088	245	0.032877	-0.1103		245	0.585826	-0.09
246	0.013511	-0.007475	246	0.032941	-0.110486		246	0.595826	-0.09
247	0.013715	-0.008248	247	0.033024	-0.110658		247	0.605826	-0.09
248	0.013919	-0.009021	248	0.033107	-0.11083		248	0.615826	-0.09
249	0.014122	-0.009795	249	0.033212	-0.110989		249	0.625826	-0.09
250	0.014326	-0.010569	250	0.033317	-0.111148		250	0.635826	-0.09
251	0.014529	-0.011342	251	0.033443	-0.111292		251	0.645826	-0.09
252	0.014732	-0.012116	252	0.033568	-0.111436		252	0.655826	-0.09
253	0.014935	-0.01289	253	0.033712	-0.111561		253	0.666667	-0.09
254	0.015137	-0.013664	254	0.033856	-0.111687				
255	0.015339	-0.014438	255	0.034015	-0.111792				
256	0.015542	-0.015212	256	0.034174	-0.111897				
257	0.015744	-0.015986	257	0.034346	-0.11198				
258	0.015947	-0.01676	258	0.034518	-0.112063				
259	0.016149	-0.017534	259	0.0347	-0.112122				
260	0.016351	-0.018308	260	0.034881	-0.112181				
261	0.016554	-0.019082	261	0.035069	-0.112215				
262	0.016756	-0.019856	262	0.035257	-0.112249				
263	0.016958	-0.02063	263	0.035448	-0.112258				
264	0.017161	-0.021404	264	0.035639	-0.112266				
265	0.017362	-0.022178	265	0.035829	-0.112249				
266	0.017563	-0.022952	266	0.036019	-0.112232				
267	0.017763	-0.023727	267	0.036391	-0.112146				
268	0.017964	-0.024501	268	0.036748	-0.112012				
269	0.018165	-0.025276							
270	0.018365	-0.02605							
271	0.018566	-0.026824							
272	0.018766	-0.027599							
273	0.018967	-0.028373							
274	0.019167	-0.029148							
275	0.019368	-0.029922							
276	0.019568	-0.030697							
277	0.019769	-0.031471							
278	0.01997	-0.032246							
279	0.02017	-0.03302							
280	0.020371	-0.033795							
281	0.020571	-0.034569							
282	0.020772	-0.035343							
283	0.020972	-0.036118							
284	0.021173	-0.036892							
285	0.021373	-0.037667							
286	0.021574	-0.038441							
287	0.021774	-0.039216							
288	0.021975	-0.03999							
289	0.022176	-0.040765							
290	0.022376	-0.041539							
291	0.022574	-0.042314							
292	0.022773	-0.043089							
293	0.022971	-0.043864							
294	0.023169	-0.044639							
295	0.023368	-0.045414							

296	0.023566	-0.046189
297	0.023764	-0.046964
298	0.023963	-0.047739
299	0.024161	-0.048514
300	0.024359	-0.049289
301	0.024556	-0.050064
302	0.024752	-0.05084
303	0.024948	-0.051615
304	0.025144	-0.052391
305	0.02534	-0.053167
306	0.025536	-0.053942
307	0.025732	-0.054717
308	0.025923	-0.055494
309	0.026114	-0.056271
310	0.026305	-0.057047
311	0.026496	-0.057824
312	0.026687	-0.058601
313	0.026877	-0.059378
314	0.027066	-0.060155
315	0.027256	-0.060933
316	0.027445	-0.06171
317	0.027635	-0.062487
318	0.027824	-0.063264
319	0.028012	-0.064042
320	0.028199	-0.064819
321	0.028387	-0.065597
322	0.028574	-0.066375
323	0.028761	-0.067153
324	0.028949	-0.06793
325	0.029136	-0.068708
326	0.029323	-0.069486
327	0.029511	-0.070264
328	0.029698	-0.071041
329	0.029885	-0.071819
330	0.030071	-0.072597
331	0.030255	-0.073375
332	0.030438	-0.074154
333	0.030622	-0.074932
334	0.030806	-0.075711
335	0.03099	-0.07649
336	0.031173	-0.077268
337	0.031357	-0.078047
338	0.031541	-0.078826
339	0.031725	-0.079604
340	0.031909	-0.080383
341	0.032092	-0.081161
342	0.032274	-0.08194
343	0.032455	-0.082719
344	0.032636	-0.083498
345	0.032817	-0.084278
346	0.032998	-0.085057

347	0.03318	-0.085836
348	0.03336	-0.086615
349	0.033541	-0.087395
350	0.033721	-0.088174
351	0.033902	-0.088954
352	0.034082	-0.089733
353	0.034262	-0.090512
354	0.03444	-0.091292
355	0.034618	-0.092072
356	0.034797	-0.092852
357	0.034975	-0.093632
358	0.035153	-0.094411
359	0.03533	-0.095191
360	0.035508	-0.095972
361	0.035685	-0.096752
362	0.035863	-0.097531
363	0.036039	-0.098312
364	0.036215	-0.099092
365	0.03639	-0.099873
366	0.036566	-0.100653
367	0.036741	-0.101434
368	0.036916	-0.102214
369	0.03709	-0.102995
370	0.037265	-0.103776
371	0.037439	-0.104556
372	0.037614	-0.105337
373	0.037787	-0.106118
374	0.037961	-0.106898
375	0.038044	-0.107289
376	0.038127	-0.107679
377	0.038204	-0.10807
378	0.038281	-0.108461
379	0.038343	-0.108849
380	0.038404	-0.109236
381	0.038404	-0.109427
382	0.038404	-0.109618
383	0.038378	-0.109807
384	0.038353	-0.109996
385	0.038303	-0.110181
386	0.038252	-0.110365
387	0.038177	-0.11054
388	0.038102	-0.110716
389	0.038004	-0.11088
390	0.037907	-0.111044
391	0.037788	-0.111193
392	0.037669	-0.111342
393	0.037531	-0.111474
394	0.037393	-0.111606
395	0.037239	-0.111718
396	0.037084	-0.111831
397	0.036748	-0.112012

APPENDIX C: NAVIER-STOKES EQUATIONS

The conservative flux vector form of the Quasi-3D Navier-Stokes equations is:

$$\frac{\partial(h\boldsymbol{Q})}{\partial t} + div(h\boldsymbol{E}(\boldsymbol{Q})) = div(h\boldsymbol{F}(\boldsymbol{Q})) + \boldsymbol{H}(\boldsymbol{Q}) \tag{C.1}$$

where the conserved variables $\boldsymbol{Q}$, the inviscid Euler flux vector $\boldsymbol{E}(\boldsymbol{Q})$, the viscous flux vector $\boldsymbol{F}(\boldsymbol{Q})$ and the source term $\boldsymbol{H}(\boldsymbol{Q})$ are defined as:

$$\boldsymbol{Q} = \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ \rho e_0 \end{Bmatrix} \qquad \boldsymbol{E}(\boldsymbol{Q}) = \begin{Bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(\rho e_0 + p) \end{Bmatrix} i + \begin{Bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(\rho e_0 + p) \end{Bmatrix} j \tag{C.2}$$

$$\boldsymbol{F}_v(\boldsymbol{Q}) = \begin{Bmatrix} 0 \\ (\tau_{xx} + \tau'_{xx}) \\ \tau_{xy} \\ u(\tau_{xx} + \tau'_{xx}) + v\tau_{xy} - \dot{q}_x \end{Bmatrix} i + \begin{Bmatrix} 0 \\ \tau_{yx} \\ (\tau_{yy} + \tau'_{yy}) \\ u\tau_{yx} + v(\tau_{yy} + \tau'_{yy}) - \dot{q}_y \end{Bmatrix} j \tag{C.3}$$

and

$$\boldsymbol{H}(\boldsymbol{Q}) = \begin{Bmatrix} 0 \\ (p - \tau'_{\theta\theta})\dfrac{dh}{dx} \\ 0 \\ 0 \end{Bmatrix} \tag{C.4}$$

For the purposes of discretisation, the source term $\boldsymbol{H(Q)}$ can be split into its inviscid and viscous components $\boldsymbol{H_i(Q)}$ and $\boldsymbol{H_v(Q)}$:

$$\boldsymbol{H}_i(\boldsymbol{Q}) = \begin{Bmatrix} 0 \\ p \\ 0 \\ 0 \end{Bmatrix} \qquad \boldsymbol{H}_v(\boldsymbol{Q}) = \begin{Bmatrix} 0 \\ -\tau'_{\theta\theta} \\ 0 \\ 0 \end{Bmatrix} \tag{C.5}$$

The removal of the $\dfrac{dh}{dx}$ term is accounted for by the discretisation.

The viscous stresses are:

$$\tau_{xx} = 2\mu\left(u_x - \frac{(u_x + v_y)}{3}\right)$$

$$\tau_{yy} = 2\mu\left(v_y - \left(\frac{u_x + v_y}{3}\right)\right)$$

$$\tau_{xy} = \tau_{yx} = \mu(u_y + v_x)$$

$$\tau'_{xx} = \tau'_{yy} = -\frac{2}{3}\mu\frac{u}{h}\frac{dh}{dx}$$

$$\tau'_{\theta\theta} = -\frac{2}{3}\mu\left(v_y + u_x - 2\frac{u}{h}\frac{dh}{dx}\right)$$

$$\text{(C.6)}$$

where

$$\dot{q}_x = \left(\frac{\mu_{lam}}{\Pr} + \frac{\mu_{turb}}{\Pr_{turb}}\right)\frac{\gamma R}{(\gamma - 1)}T_x$$

$$\dot{q}_y = \left(\frac{\mu_{lam}}{\Pr} + \frac{\mu_{turb}}{\Pr_{turb}}\right)\frac{\gamma R}{(\gamma - 1)}T_y$$

$$\text{(C.7)}$$

Variables ρ, u and v are the density and the velocity components in the x and y directions, e_0 is the total energy per unit volume, k_c is the thermal conductivity, p is the static pressure, and h is the stream-tube thickness. The additional terms τ'_{xx} and $\tau'_{\theta\theta}$ are associated with variations in stream-tube thickness and would not be present in a 2D implementation. Subscripts x and y denote partial derivatives in the respective directions.

$\dot{q}_x$ and $\dot{q}_y$ are the components of the heat flux in the x and y directions, modelled according to Fourier's Law, and $\Pr$ and $\Pr_{turb}$ are the laminar and turbulent Prandtl numbers respectively.

The equations are closed with the pressure equation:

$$p = (\gamma - 1)\left\{\rho e_0 - \frac{\rho(u^2 + v^2)}{2}\right\}$$

$$\text{(C.8)}$$

where γ is the ratio of specific heats $\dfrac{C_P}{C_V}$ and is generally taken as 1.4 for air.

μ is the sum of the laminar and turbulent viscosities, $\mu = \mu_{lam} + \mu_{turb}$ The laminar viscosity is calculated using Sutherland's Law (valid for temperatures $\geq 110.4°K$):

$$\mu_{lam} = 1.458 \times 10^{-6} \cdot \frac{T^{3/2}}{(T + 110.4)} \tag{C.9}$$

μ_{turb} is reserved for the turbulence model in question.

The thermal conductivity k_c is derived from the Prandtl number:

$$Pr = \frac{\mu C_p}{k_c} \tag{C.10}$$

The ratio $\dfrac{C_p}{Pr}$ is approximately constant for most gases, so once μ has been evaluated, the thermal conductivity can be quickly calculated. Pr has been taken as 0.72, $C_p = 1006.43$ J/kg.K, T is static temperature and R is the specific gas constant taken as 287.1 J/kg.K

Non-Dimensionalisation

Non-dimensionalisation is performed to simplify the act of comparing flows that are considered to be dynamically similar. This is achieved by replacing each of the dimensional variables by the product of a constant reference parameter and a non-dimensional component. Freestream conditions are used to non-dimensionalise the flow:

$$
\begin{aligned}
t^* = \frac{tu_\infty}{L} \qquad & x^* = \frac{x}{L} \qquad & y^* = \frac{y}{L} \\[2mm]
\mu^* = \frac{\mu}{\mu_\infty} \qquad & u^* = \frac{u}{u_\infty} \qquad & v^* = \frac{v}{v_\infty} \\[2mm]
\rho^* = \frac{\rho}{\rho_\infty} \qquad T^* = \frac{T}{T_\infty} \qquad & p^* = \frac{p}{\rho_\infty u_\infty^2} \qquad & e_0^* = \frac{e_0}{\rho_\infty u_\infty^2}
\end{aligned}
\tag{C.11}
$$

The conservative form of the Navier-Stokes equations can then be defined as:

$$\frac{\partial (h\boldsymbol{Q})}{\partial t} + div(h\boldsymbol{E}(\boldsymbol{Q})) = \frac{1}{Re} div(h\boldsymbol{F}(\boldsymbol{Q})) + \boldsymbol{H}_i(\boldsymbol{Q}) + \frac{1}{Re}\boldsymbol{H}_v(\boldsymbol{Q}) \tag{C.12}$$

The important parameters are now reduced to two, namely the Reynolds number $Re = \dfrac{\rho_\infty u_\infty L_\infty}{\mu_\infty}$ and the Mach number $M_\infty = \dfrac{U_\infty}{\sqrt{\gamma R T_\infty}}$.

This is discretised in the following manner:

$$\frac{\partial \boldsymbol{Q}_n}{\partial t} = -\frac{1}{2 A_n h_n} \sum_{k=1}^{d_n} \left\{ \begin{array}{l} \overline{h}_k \left[\boldsymbol{E}(\boldsymbol{Q}_{n,k}) \cdot \hat{n} - 2\boldsymbol{H}_i(\boldsymbol{Q}_n)\hat{n}_x - \dfrac{2}{Re}\boldsymbol{H}_v(\boldsymbol{Q}_n)\hat{n}_x \right] \\[2ex] -\dfrac{\overline{h}_e}{Re}\left[\boldsymbol{F}(\boldsymbol{Q}_{n,k,k^+}) \cdot \hat{v} \right] \end{array} \right\} \tag{C.13}$$

where $\hat{n}$ is the normal vector to Face k and equals the length of Face k.

$\hat{v}$ is the normal vector to edge $k \to k^+$ and is the length of that edge.

$\|\hat{n}\|$ represents the length of Face k.

A_n is the area of cell Ω_n.

$\boldsymbol{F}(\boldsymbol{Q}_{n,k,k+})$ is the element averaged viscous flux term at the centre of the triangle (n,k,k^+).

$\boldsymbol{H}_i(\boldsymbol{Q}_n)$ and $\boldsymbol{H}_v(\boldsymbol{Q}_n)$ are the inviscid and viscous source terms evaluated at node n.

$\overline{h}_k$ is the stream-tube thickness at the centre of the edge $n \to k$.

$\overline{h}_e$ is the stream-tube thickness at the centre of the triangle (n,k,k^+).

REFERENCES

Allen, C.B. (1995). "Central-Differencing and Upwind Biased Schemes for Steady and Unsteady Euler Aerofoil Computations". *Aeronautical Journal*. No. 2, pp. 52-62.

Anderson, W.K. (1992). "Grid Generation and Flow Solution Method for the Euler Equations on Unstructured Grids". *NASA Technical Memorandum*, NASA TM-4295.

Andrew, P.L., and Ng, W. (1994). "Turbulence Characteristics in a Supersonic Cascade Wake Flow". *Transactions of the ASME*. **116**, pp. 586-596.

Arnone, A., and Pacciani, R. (1997). "Numerical Prediction of Trailing Edge Wake Shedding". *ASME Conference Papers,* 97-GT-89.

Arnone, A., Liou, M., and Povinelli, L.A. (1995). "Integration of Navier-Stokes Equations Using Dual Time Stepping and a Multigrid Method". *AIAA Journal*. **33**, No.6, pp. 985-990.

Bank, R.E. (1985). *PLTMG user's guide.* Technical Report Edition 4, University of California, San Diego.

Barth, T.J. (1991). "Numerical Aspects of Computing Viscous High Reynolds Number Flows on Unstructured Meshes". *AIAA Conference Papers*. AIAA-91-0721.

Barth, T.J., and Jesperson, D.C. (1989). "The Design and Application of Upwind Schemes on Unstructured Meshes". *AIAA Conference Papers*. AIAA-89-0366.

Batina, J.T. (1993). "Implicit Upwind Solution Algorithms for Three-Dimensional Unstructured Meshes". *AIAA Journal*. **31**, No. 5, pp. 801-805.

Biswas, D., and Fukuyama, Y. (1996). "Calculation of Transition Boundary Layers with an Improved Low-Reynolds-Number Version of the k-ε Turbulence Model". *Journal of Turbomachinery*. **116**, pp. 765-773.

Bowyer, A. (1981). "Computing Dirichlet Tessellations". *The Computer Journal*. **24**, No. 2, pp. 162-166.

Braaten, M.E., and Connell, S.D. (1996). "Three-Dimensional Unstructured Adaptive Multigrid Scheme for the Navier-Stokes Equations". *AIAA Journal*. **34**, No. 2, pp. 281-290.

Bryanston-Cross, P.J., and Camus, J.J. (1982). "Auto and Cross-correlation Measurements in a Turbine Cascade Using a Digital Correlater". *ASME Conference Papers,* 82-GT-132.

Bryanston-Cross, P.J., and Oldfield, M.L.G. (1981). "Interferometric Measurements in a Turbine Cascade using Image-plane Holography". *ASME Journal*, **103**, No. 1, pp. 124-130.

Carscallen, W.E., Currie, T.C., Hogg, S.I., and Gostelow, J.P. (1999). "Measurement and Computation of Energy Separation in the Vortical Wake Flow of a Turbine Nozzle Cascade". *Journal of Turbomachinery.* **121**, No. 4, pp. 703-708.

Carscallen, W.E., Hogg, S.I., Gostelow, J.P., and Buttsworth, D.R. (1997) "Time Resolved Measurements of the Energy Separation Process in a Transonic Turbine Vane Wake Flow". *Advanced Aerodynamic Measurement Technology.* AGARD-CP-601, Sept. 22-25.

Carscallen, W.E., Fleige, H., and Gostelow, J.P. (1996). "Transonic Turbine Vane Wake Flows". *ASME Conference Papers,* 96-GT-419.

Carscallen, W.E., and Gostelow, J.P. (1994). "Observations of Vortex Shedding in the Wake from Transonic Turbine Nozzle Blades". *ISROMAC-5*, **A**, May 8-11.

Carscallen, W.E., and Oosthuizen, P.O. (1989). "The Effect of Secondary Flow on the Redistribution of the Total Temperature Field Downstream of a Stationary Turbine Cascade". *AGARD Conference Papers,* AGARD-CP-469.

Cicatelli, G., and Sieverding, C.H. (1996) "The Effect of Vortex Shedding on the Unsteady Pressure Distribution Around the Trailing Edge of a Turbine Blade". *ASME Conference Papers,* 96-GT-359.

Cicatelli, G., and Sieverding, C.H. (1996) "A Review of the Research on Unsteady Turbine Blade Wake Characteristics". *AGARD Conference Papers*, AGARD-CP-57, Paper 6, Jan.

Connell, S.D., Holmes, D.G., and Braatan, M.E. (1993). "Adaptive Unstructured 2D Navier-Stokes Solutions on Mixed Quadrilateral/Triangular Meshes". *ASME Conference Papers,* 93-GT-99.

Currie, T.C., and Carscallen, W.E. (1998). "Simulation of Trailing Edge Vortex Shedding in a Transonic Turbine Cascade". *Journal of Turbomachinery.* **120**, No. 1, pp. 10-19.

Cuthill, E., and McKee, J. (1969). "Reducing the bandwidth of sparse symmetric matrices". *ACM Publications.* Proceedings of the 24th ACM National Conference, pp. 157-172.

Dadone, A., and Grossman, B. (1992). "Characteristic Based, Rotated Upwind Scheme for the Euler Equations". *AIAA Journal.* **30**, No. 9, pp. 2219-2226.

Denton, J.D. (1993). "Loss Mechanisms in Turbomachines". *Journal of Turbomachinery.* **115**, No. 4, pp 621-656.

Donat, R., and Marquina, A. (1995). "Capturing Shock Reflections: An Improved Flux Formula". *Journal of Computational Physics.* **125**, pp. 42-58.

Douglas, J.F., Gasiorek, J.M., and Swaffield, J.A. (1995). "Fluid Mechanics: 3rd Edition". *Longman Scientific and Technical Publishers.*

Dwyer, R.A. (1987). "A Faster Divide-and-conquer Algorithm for Constructing Delaunay Triangulations". *Algorithmica.* **2**, pp. 137-151.

Eckert, E.R.G. (1984). "Experiments on Energy Separation in Fluid Streams". *Mechanical Engineering*. **106**, No. 10, pp. 58-65.

Eckert, E.R.G. (1986). "Energy Separation in Fluid Streams". *International Committee on Heat and Mass Transfer*. **13**, pp. 127-143.

Faber, T.E. (1995). "Fluid Dynamics for Physicists". *Cambridge University Press*.

Foley, J., Van Dam, A., Feiner, S., and Hughes, J. (1990). "Computer Graphics, Principles and Practice". *Addison-Wesley Publishing Company*.

Fox, M.D., Kurosaka, M., Hedges, L., and Hirano, K. (1993). "The Influence of Vortical Structures on the Thermal Fields of Jets". *Journal of Fluid Mechanics*. **255**, pp. 447-472.

Frink, N.T. (1992). "Upwind Scheme for Solving the Euler Equations on Unstructured Tetrahedral Meshes". *AIAA Journal*. **30**, No. 1, pp. 70-77.

Gehrer, A., Lang, H., Mayrhofer, N., and Woisetschläger. (2000). "Numerical and Experimental Investigation of Trailing Edge Vortex Shedding Downstream of a Linear Turbine Cascade". *ASME Conference Papers*, 2000-GT-434.

Godunov, S.K. (1959). "Finite Difference Method for Numerical Computation of Discontinuous Solutions of the Equations of Fluid Dynamics," *Matematicheskii Sbornik*. **47**, No. 3, pp. 271-306.

Gostelow, J.P. (1984). "Cascade Aerodynamics". *Pergamon Press*.

Graf, W.H. (1998). "Fluvial Hydraulics: Flow and Transport Porcesses in Channels of Simple Geometry". *John Wiley & Sons Publishers*.

Guibas, L., and Stolfi, J. (1985). "Primitives for the Manipulation of General Subdivisions and the Computation of Voronoi Diagrams". *ACM Trans. Graphics*. **4**, pp. 74-123.

Guibas, L.J., Knuth, D., and Sharir, M. (1992). "Randomized Incremental Construction of Delaunay and Voronoi Diagrams". *Algorithmica*. **7**, pp. 381-413.

Gustafsson, B., and Sundström, A. (1978). "Incomplete Parabolic Problems in Fluid Dynamics". *SIAM Journal of Applied Mathematics*. **35**, pp. 343-357.

Gutsol, A.F. (1997). "The Ranque Effect". *Uspekhi Fizicheskikh Nauk*, Russian Academy of Sciences. **40**, No. 6, pp. 639-658.

Hall, K.C., and Clark, W.S. (1993). "Linearised Euler Predictions of Unsteady Aerodynamic Loads in Cascades". *AIAA Journal*. **31**, No. 3, pp. 540-550.

Harten, A. (1983). "High Resolution Schemes for Hyperbolic Conservation Laws". *Journal of Computational Physics*. **49**, pp. 357-393.

Harten, A. (1984). "On a Class of High Resolution Total Variation Stable Finite Difference Schemes". *SIAM Journal of Numerical Analysis*. **21**, pp. 1-23.

Harten, A., and Hyman, J.M. (1983). "Self Adjusting Grid Methods for One Dimensional Hyperbolic Conservation Laws". *Journal of Computational Physics*. **50**, pp. 235-269.

Heinemann, J., and Butefish, K.A. (1977). "Determination of the Vortex Shedding Frequency of Cascades with Different Trailing Edge Thicknesses". *AGARD Conference Papers, AGARD-CP-227*, Paper 1.

Hirsch, C. (1990). "Numerical Computation of Internal and External Flows. II. Computational Methods for Inviscid and Viscous Flows". *Wiley Series in Numerical Methods in Engineering*. **2**.

Hixon, R. (1999). "Prefactored Compact Filters for Computational Aeroacoustics". *AIAA Conference Papers*. AIAA-99-0358.

Hoffman, K.A., and Chiang, S.T. (1993). "Computational Fluid Dynamics for Engineers". *Engineering Education Systems*. **1&2**.

Jameson, A., Schmidt, W., and Turkel, E. (1981). "Numerical Simulation of the Euler Equations by Finite Volume Methods using Runge-Kutta Time-stepping Schemes." *AIAA Conference Papers*. AIAA-81-1259.

Kurosaka, M., Gertz, J.B., Graham, J.E., Goodman, J.R., Sundaram, P., Riner, W.C., Kuroda, H., and Hankey, W.L. (1987). "Energy Separation in a Vortex Street". *Journal of Fluid Mechanics*. **178**, pp. 1-29.

Kurosaka, M., Graham, J.E., and Shang, J.S. (1989). "Negative Heat Transfer in Separated Flows". *International Journal of Heat and Mass Transfer*. **32**, No. 6, pp. 1192-1195.

Lawson, C.L. (1977). "Software for C^1 Surface Interpolation". *Mathematical Software III*, editor: Rice, J.R., pp. 161-194. Academic Press.

Levy, D. (1990). "Use of a Rotated Riemann Solver for the Two Dimensional Euler Equations". *PhD Thesis*, University of Michigan.

Loth, E., Baum, J., and Löhner, R. (1992). "Formation of Shocks Within Axisymmetric Nozzles". *AIAA Journal*. **20**, No. 1, pp. 268-270.

Luo, H., Baum, J.D., and Löhner, R. (1994). "Edge-Based Finite Element Scheme for the Euler Equations". *AIAA Journal*. **32**, No. 6, pp. 1183-1190.

Manna, M. (1992). "A Three Dimensional High Resolution Compressible Flow Solver". *PhD Thesis*. Von Karman Institute for Fluid Dynamics.

Mavriplis, D.J. (1993). "An Advancing-Front Delaunay Triangulation Algorithm Designed for Robustness". *AIAA Conference Papers*. AIAA-93-0671.

Menter, F.R. (1993). "Zonal Two-Equation k-ω Turbulence Models for Aerodynamic Flows". *AIAA Conference Papers*. AIAA-93-2906.

Merriam, M.L. (1991). "An Efficient Advancing Front Algorithm for Delaunay Triangulation". *AIAA Conference Papers*. AIAA-91-0792.

Morton, K.W. (1996). "Numerical Solution of Convection-Diffusion Problems". *Applied Mathematics and Mathematical Computation*, **12**. Chapman & Hall publishers.

Moustapha, S.H., Carscallen, W.E., and McGeachy, J.D. (1993). "Aerodynamic Performance of a Transonic Low Aspect Ratio Turbine Nozzle". *Transactions of the ASME*. **115**, No. 7, pp. 400-408.

Ng, W.F., Chakroun, W.M., and Kurosaka, M. (1990). "Time-Resolved Measurements of Total Temperature and Pressure in the Vortex Street behind a Cylinder". *Physics of Fluids A*. **2**, No. 6, pp. 971-979.

Ni, R.H., and Bogoian, J. (1989). "Prediction of 3D Multi-stage Turbine Flow Field Using a Multiple Grid Euler Solver". *AIAA Conference Papers*. AIAA-89-0203.

Ollivier-Gooch, C.F. (1995). "Multigrid Acceleration of an Upwind Euler Solver on Unstructured Meshes". *AIAA Journal*. **33**, No. 10, pp. 1822-1827.

Pan, D., and Cheng, J. (1993). "Upwind Finite-Volume Navier-Stokes Computations on Unstructured Triangular Meshes". *AIAA Journal*. **31**, No. 9, pp. 1618-1625.

Quirk, J.J. (1991). "An Adaptive Grid Algorithm for Computational Shock Hydrodynamics". *PhD Thesis*. Cranfield Institute of Technology, College of Aeronautics.

Roberts, Q.D., and Denton, J.D. (1996). "Loss Production in the Wake of a Simulated Subsonic Turbine Blade". *ASME Conference Papers,* 96-GT-421.

Roe, P.L. (1981). "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes". *Journal of Computational Physics*. **43**, pp. 357-372.

Rumsey, C. (1991). "Effect of a multi-dimensional flux function on the monotonicity of Euler and Navier-Stokes equations". *AIAA Conference Papers*. AIAA-91-1530.

Rumsey, C., Van Leer, B., and Roe, P.L. (1991). "A grid independent approximate Riemann solver with applications to the Euler and Navier-Stokes equations". *AIAA Conferences Papers*. AIAA-91-0239.

Saad, Y., and Schultz, M.H. (1986). "GMRES: A Generalised Minimal Residual Algorithm for Solving Non-Symmetric Linear Systems". *Journal of Scientific and Statistical Computing*. **7**, No. 3, pp. 856-869.

Saiva, G. (1962). "Hydraulic Analogy Studies of Shock Attachments to Wedges". *Proceedings of the 1st Australian Conference on Hydraulics and Fluid Dynamics*.

Sanz, W., and Platzer, M.F. (1998). "On the Navier-Stokes Calculation of Separation Bubbles with a new Transition Model". *Journal of Turbomachinery*. **120**, No. 1, pp. 36-42.

Shepherd, D.G. (1965). "Elements of Fluid Mechanics". *Harcourt, Brace and World Publishers*.

Shewchuk, J.R. (1996). "Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator". *First Workshop on Applied Computational Geometry*. May, pp. 124-133. Association for Computing Machinery.

Sieverding, C.H., and Heinemann, J. (1990). "The Influence of Boundary Layer State on the Vortex Shedding from Flat Plates and Turbine Cascades". *Journal of Turbomachinery*. **112**, No. 4, pp. 181-187.

Smith, H.E. (1967). "The Flow Field and Heat Transfer Downstream of a Rearward Facing Step in Supersonic Flow". *American Research Libraries*. ARL-67-0056.

Steger, J.L., and Warming, R.F. (1981). "Flux Vector Splitting of the Inviscid Gasdynamic Equations with Applications to Finite-Difference Methods". *Journal of Computational Physics*. **40**, pp. 263-293.

Sweby, P.K. (1984). "High Resolution Schemes using Flux Limiters for Hyperbolic Conservative Laws". *SIAM Journal of Numerical Analysis*. **21**, pp. 995-1011.

Tanemura, M., Ogawa, T., and Ogita, N. (1983). "A New Algorithm for Three-Dimensional Voronoi Tessellation". *Journal of Computational Physics*. **51**, pp. 191-207.

Toro, E.F. (1999). "Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction". *Springer publications*.

Trépanier, J.Y., Reggio, M., Paraschivoiu, M., and Camarero, R. (1993). "Unsteady Euler Solutions for Arbitrary Moving Bodies and Boundaries". *AIAA Journal*. **31**, No. 10, pp. 1869-1876.

Van Albada, G.D., Van Leer, B., and Roberts, W.W. (1982). "A Comparative Study Of Computational Methods In Cosmic Gas Dynamics". *Astron. Astrophysics*. **108**, pp. 76-84.

Van Leer, B. (1973). "Towards the Ultimate Conservative Difference Scheme. I. The Quest of Monotonicity". *Lecture Notes in Physics*. **18,** pp. 163-168. Springer Verlag. Berlin.

Van Leer, B. (1974). "Towards the Ultimate Conservative Difference Scheme. II. Monotonicity and Conservation Combined in a Second Order Scheme". *Journal of Computational Physics*. **14**, pp. 361-370.

Van Leer, B. (1977a). "Towards the Ultimate Conservative Difference Scheme. III. Upstream-centred Finite Difference Schemes for Ideal Compressible Flow". *Journal of Computational Physics*. **23**, pp. 263-275.

Van Leer, B. (1977b). "Towards the Ultimate Conservative Difference Scheme. IV. A New Approach to Numerical Convection". *Journal of Computational Physics*. **23**, pp. 276-299.

Van Leer, B. (1979). "Towards the Ultimate Conservative Difference Scheme. V. A Second Order Sequel to Godunov's Method". *Journal of Computational Physics*. **32**, pp. 101-136.

Xu, L., and Denton, J.D. (1987). "The Base Pressure and Loss of a Family of Four Turbine Blades". *ASME Conference Papers*, 87-GT-202.

Xu, L., and Denton, J.D. (1989). "The Trailing Edge Loss of Transonic Turbine Blades". *ASME Conference Papers*, 89-GT-278.

Watson, D. (1981). "Computing the *n*-dimensional Delaunay Tessellation with Application to Voronoi Polytopes". *The Computer Journal*. **24**, No. 2, pp. 167-172.

Wilcox, D.C. (1988). "Reassessment of the Scale-Determining Equation for Advanced Turbulence Models". *AIAA Journal*. **26**, No. 11, pp. 1299-1310.

Yang, J.Y., and Hsu, C.A. (1992). "High-Resolution, Non-oscillatory Schemes for Unsteady Compressible Flows". *AIAA Journal*. **30**, No. 6, pp. 1570-1575.

Zhang, X.D., Trépanier, J.Y., Reggio, M., Benmeddour, A., and Camarero, R. (1996). "Grid Influence on Upwind Schemes for the Euler and Navier-Stokes Equations". *AIAA Journal*. **34**, No. 4, pp. 717-727.

Zheng, L. (1993). "Numerical Solutions of Navier-Stokes Equations and k-ω Turbulence Model Equations using a Staggered Upwind Method". *AIAA Conference Papers*. AIAA-93-2968.

Zheng, X., and Liu, F. (1995). "Staggered Upwind Method for Solving Navier-Stokes and k-ω Turbulence Model Equations". *AIAA Journal*. **33**, No. 6, pp. 991-998.